Programmable Device Server User Manual

Version 2.0, February 2013

Service and usage information for

DS-700

PDS-700

PPDS-700-MTCP

PDS-5105D-MTCP

PDSM-700

PPDSM-700-MTCP

PPDS-700-IP67

Warranty

All products manufactured by ICP DAS are under warranty regarding defective materials for a period of one year, starting from the date of delivery to the original purchaser.

Warning

ICP DAS assumes no liability for damages resulting from the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Trademark

The names used for identification only may be registered trademarks of their respective companies.

Contact US

If you have any question, please feel free to contact us.
We will give you quick response within 2 workdays.

Email: service@icpdas.com , service.icpdas@gmail.com

Table of Contents

Packing List

The package includes the following items:

- One (Programmable) Device Server hardware module
- One software utility CD
- One RS-232 download cable, CA-0910
  (Only for PDS(M)-700(D), PPDS(M)-700(D)-MTCP, PDS-5105D-MTCP)
- One Quick Start Guide

Note: If any of these items are missed or damaged, contact the local distributors for more information. Save the shipping materials and cartons in case you want to ship in the future.

More Information

Documentations

CD: Napdos\PDS\PDS-700\Document
http://ftp.icpdas.com/pub/cd/8000cd/napdos/pds/pds-700/document/

VxComm Driver (Virtual COM)

CD: \NAPDOS\ Driver\VxComm_Driver
http://ftp.icpdas.com/pub/cd/8000cd/napdos/driver/vxcomm_driver/

Firmware

CD:\ Napdos\PDS\PDS-700\VxComm\Server(PDS)
http://ftp.icpdas.com/pub/cd/8000cd/napdos/pds/pds-700/vxcomm/server(pds)/

MiniOS7

CD:\NAPDOS\PDS\PDS-700\OS_image
http://ftp.icpdas.com/pub/cd/8000cd/napdos/pds/pds-700/os_image/

1. Introduction

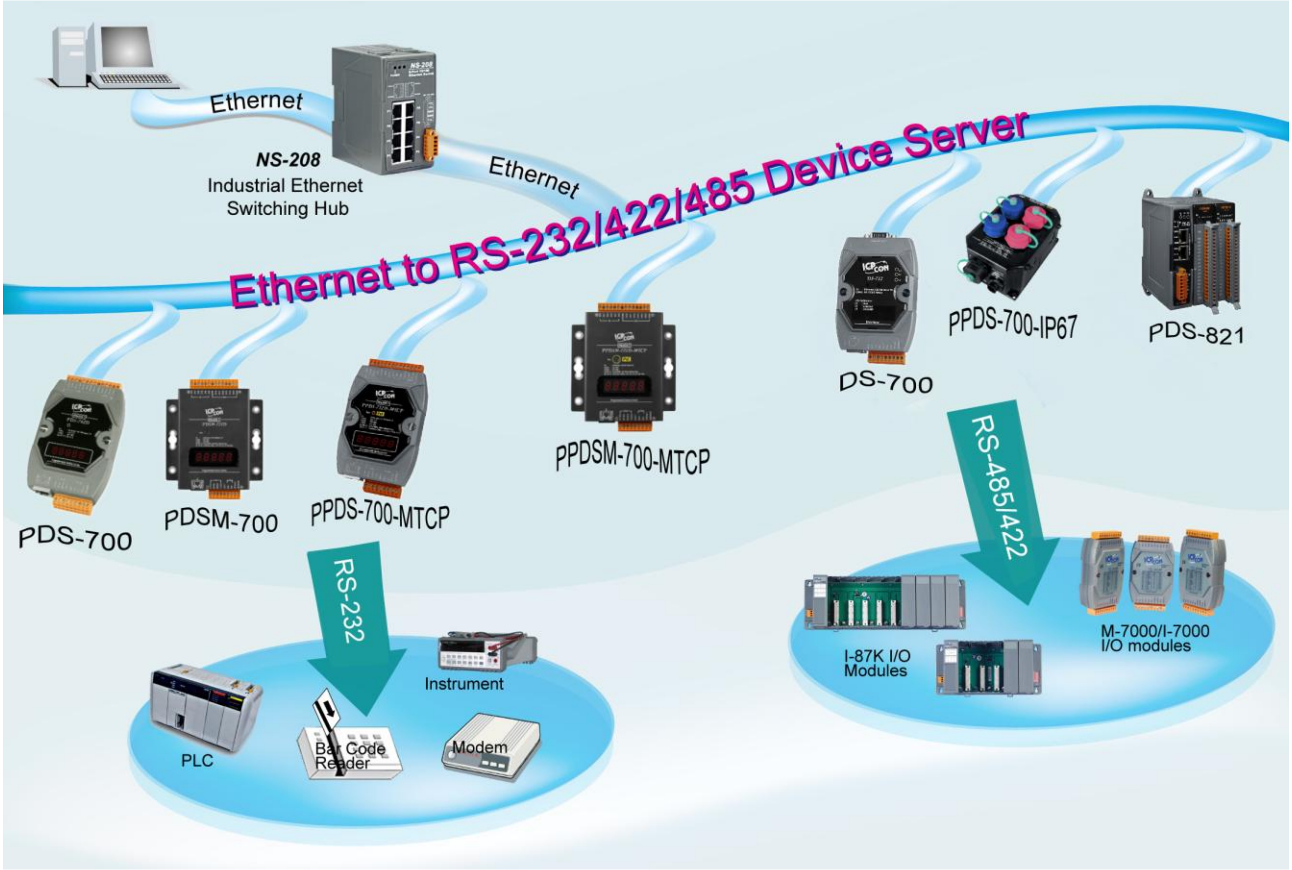
The PDS-700 series is a family of Programmable Device Servers, also known as "Serial-to-Ethernet gateway", that are designed for linking RS-232/422/485 devices to an Ethernet network. The user-friendly VxComm Driver/Utility allows users to easily turn the built-in COM ports of the PDS-700 series into standard COM ports on a PC. By virtue of its protocol independence, a small-core OS and high flexibility, the PDS-700 series is able to meet the demands of every network-enabled application.

The PDS-700 series includes a powerful and reliable Xserver programming structure that allows you to design your robust Ethernet applications in one day. The built-in, high-performance MiniOS7 boots the PDS-700 up in just one second and gives you fastest responses.

The PPDS-700-MTCP series features true IEEE 802.3af-compliant (classification, Class 1) Power over Ethernet (PoE) using a standard category 5 Ethernet cable to receive power from a PoE switch like the NS-205PSE. The PPDS-700-MTCP also works as a Modbus/TCP to Modbus/RTU gateway that supports most SCADA/HMI communications based on the Modbus/TCP protocol. The PDSM-700 is the PDS-700 with Metal Case (RoHS) and the PPDSM-700-MTCP is the PPDS-700-MTCP with Metal Case (RoHS). Metal Case version includes stronger protection than PDS-700 and PPDS-700-MTCP.

1.1 Ethernet Solutions

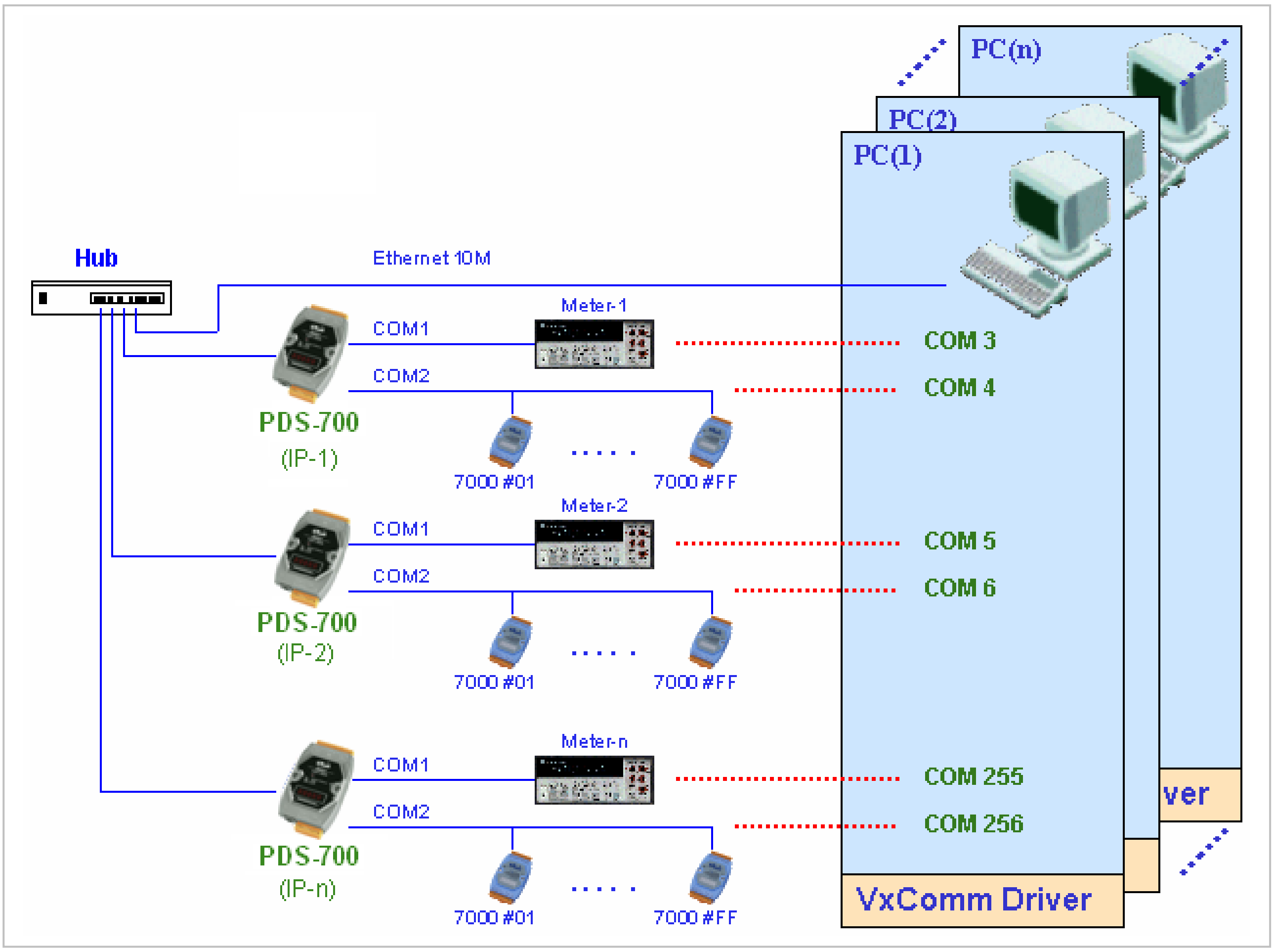
Nowadays, the Ethernet protocol has become the de-facto standard for local area networks. Via the Internet, connectivity is occurring everywhere, from home appliances, to vending machines, to testing equipment, to UPS ...etc. An Ethernet network can link office automation and industrial control networks, access remote systems and share data and information between multivendor machines; it also provides a cost-effective solution for industrial control networks.



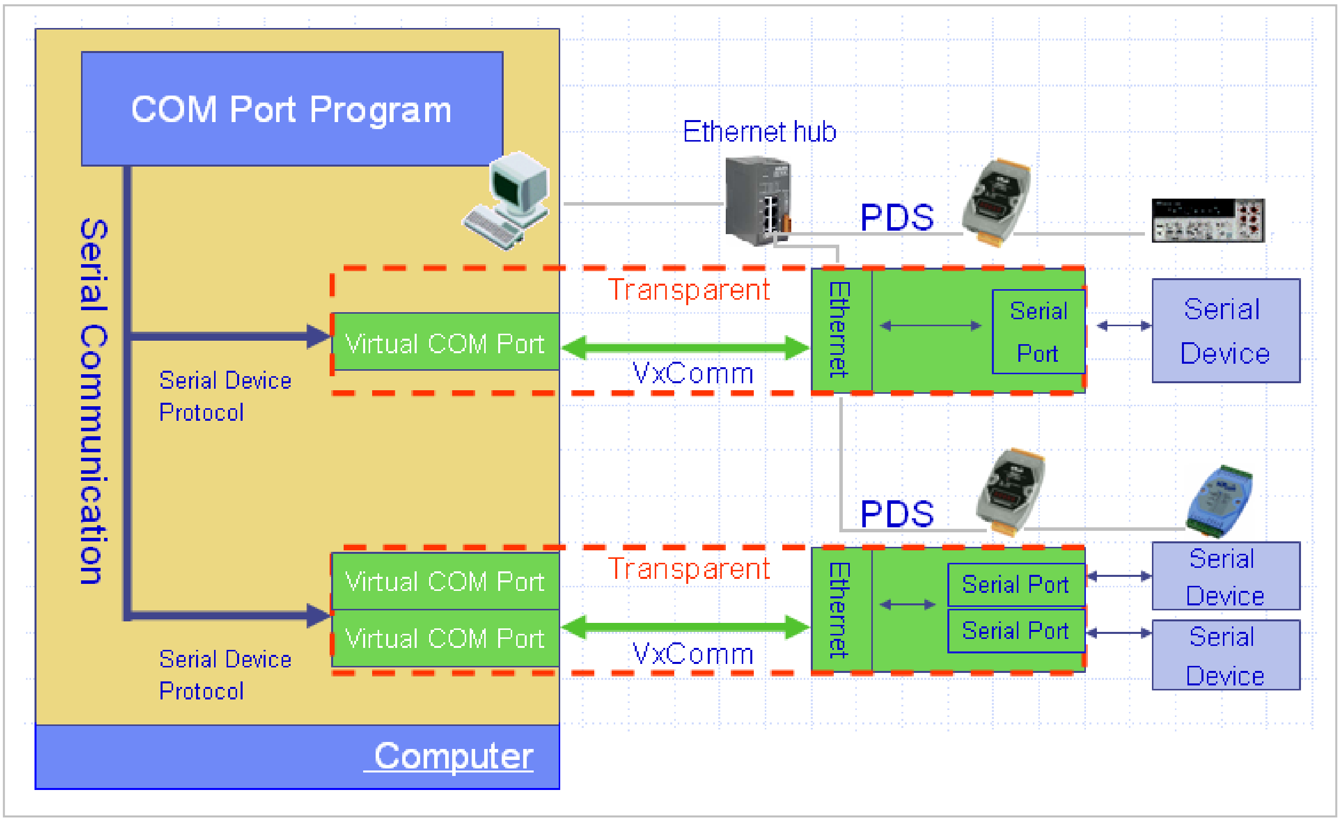
1.2 VxComm Technology

In general, writing a TCP/IP program is more difficult than a COM port program, or the COM port communication system was built many years ago.

As a result, a new technology, **VxComm** was developed to **virtualize the COM ports of the PDS to allow up to 256 COM Ports to be used on the central computer.** The VxComm driver saves time when accessing serial devices through the Ethernet without the need for reprogramming the COM port software on the PC.

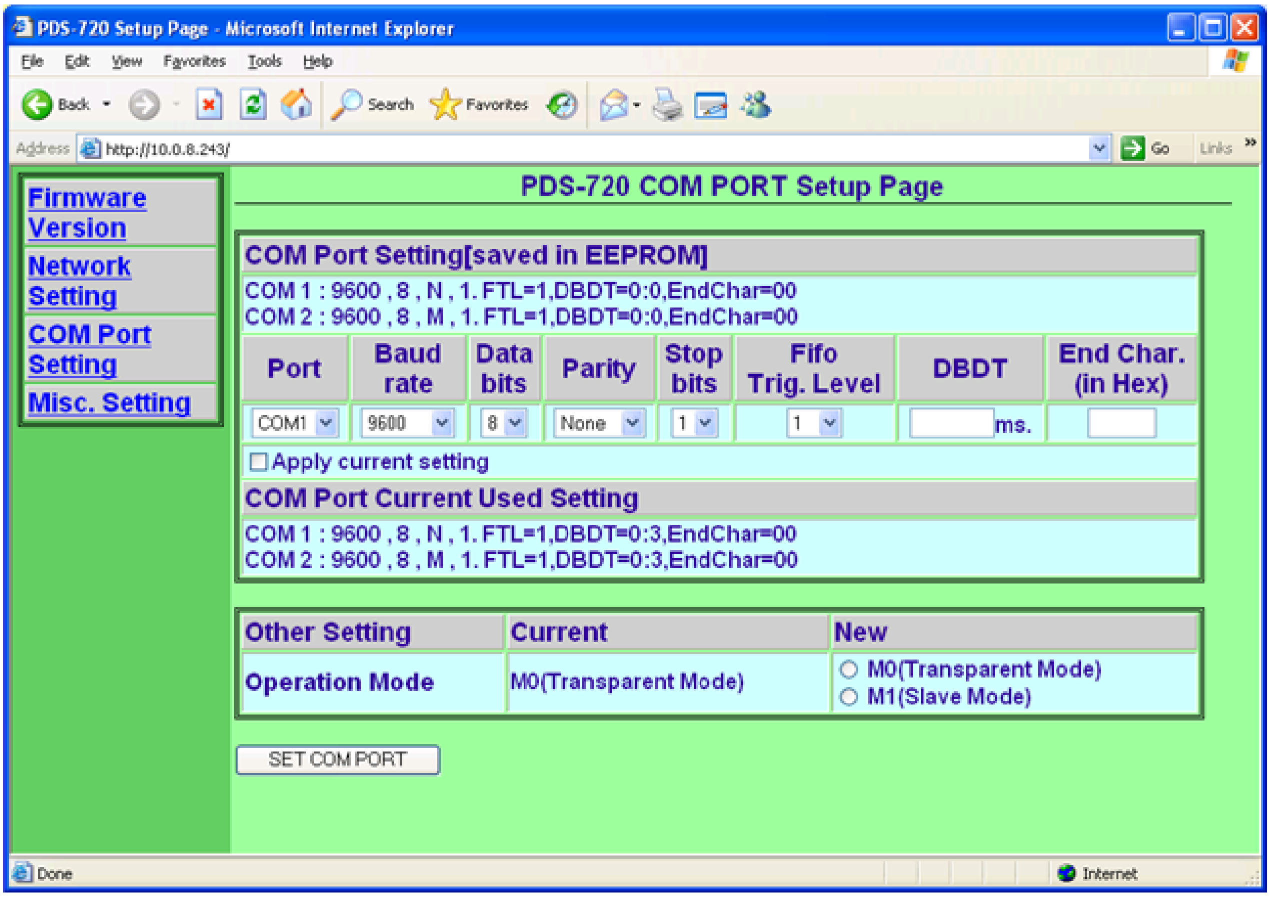


The VxComm driver controls all the details of the Ethernet TCP/IP programming technique; your COM port program will be able to access your serial devices through Ethernet in the same way as through COM port with the assistance of PDS and VxComm technology.



1.3 Web Server Technology

Web server technology enables configuration of the PDS via a standard web browser interface, e.g. Internet Explorer, FireFox or Mozilla, etc. This means that it is easy to check the configuration of the PDS via an Ethernet network without needing to install any other software tools; thereby reducing the user's learning carve.


PDS-720 Setup Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://10.0.8.243/ Go Links
Firmware Version
Network Setting
COM Port Setting
Misc. Setting
SET COM PORT
Done
Internet

PDS-720 COM PORT Setup Page

COM Port Setting[saved in EEPROM]

COM 1 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00
COM 2 : 9600 , 8 , M , 1. FTL=1,DBDT=0:0,EndChar=00

Port	Baud rate	Data bits	Parity	Stop bits	Fifo Trig. Level	DBDT	End Char. (in Hex)
COM1	9600	8	None	1	1	ms.	

□ Apply current setting

COM Port Current Used Setting

COM 1 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00
COM 2 : 9600 , 8 , M , 1. FTL=1,DBDT=0:3,EndChar=00

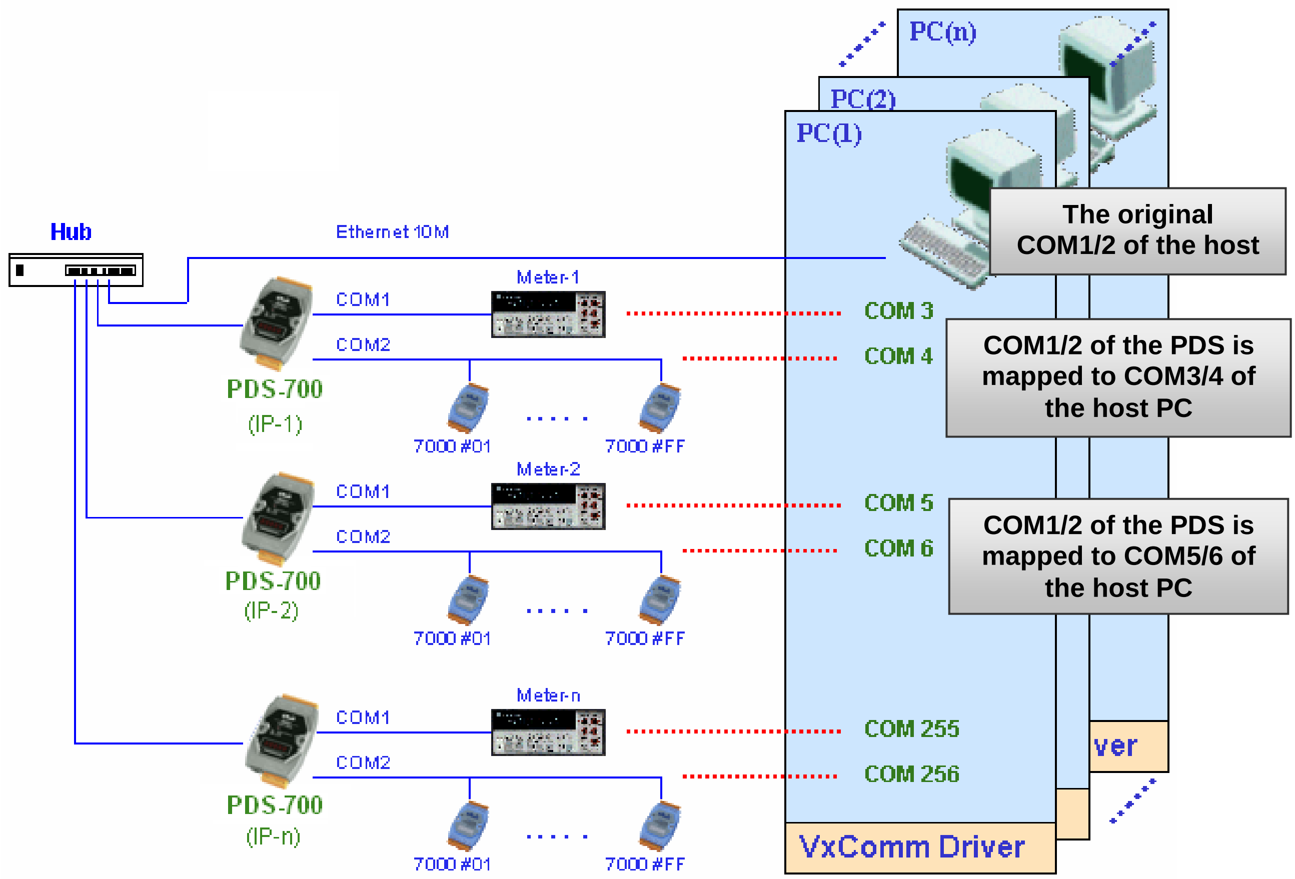
Other Setting	Current	New
Operation Mode	M0(Transparent Mode)	○ M0(Transparent Mode) ○ M1(Slave Mode)

2. Typical Applications for the PDS

2.1 RS-232/485/422 Device Networking

--- Using Virtual COM Technology ---

The PDS series is designed to link RS-232/485/422 devices to an Ethernet network. The VxComm utility allows the built-in PDS COM Port to be virtualized to a standard COM Port of the host PC as shown below:



In the configuration above, Meter-1 is virtualized to link to COM3 of the host PC. Therefore a program original designed for the MS-COMM standard can access the meter **without any modification.**

2.2 Ethernet I/O Applications

The PDS series provides 2 types of Ethernet I/O solutions:

- Linking to I-7000 series modules
- Built-in DIO (if the module supports the DIO function)

Linking to I-7000 series modules

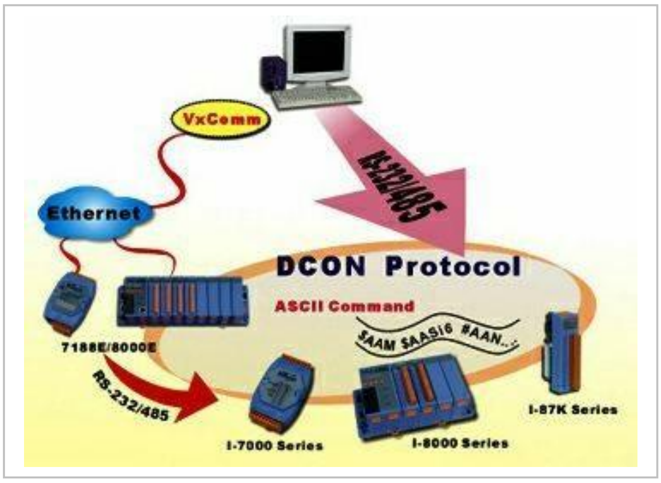
The I-7000 series provides a variety of I/O operations, such as D/I, D/O, A/D, D/A, Counter and Frequency Measurement, etc. The I-7000 series was originally designed to be used with RS-485 networks, so COM2 on the PDS-700 can be used to link to I-7000 series modules.

By using VxComm technology, programs that on the host PC support serial devices can be upgraded from a RS-485 network to an Ethernet network without requiring any modifications to the program. Refer to Sec. 2.1 for more information.

Built-in DIO

The DCON protocol is a request /reply communication. Protocol that is defined using a simple ASCII format, such as $AAN, $AASi6, #AAN, etc. and is used to access PDS and I-7000/8000/ 87k series I/O modules.

The DCON protocol command set for the PDS is introduced in Sec. 7. The protocol allows access the built-in I/O through the virtual COM Ports mapped to the Port I/O of the PDS in the VxComm Utility.



2.3 Linking I-7000 Module to an Ethernet Network

The I-7000 family was originally designed for use with an RS-485 network. They are very robust and work well under the harsh industrial environments.

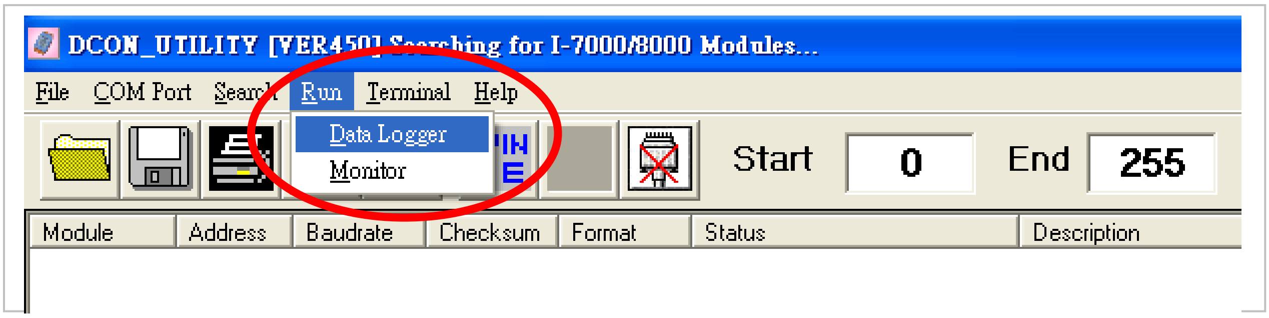
The PDS enables I-7000 modules to be upgraded to an Ethernet solution. Linking I-7000 modules to an Ethernet combines the advantages of both RS-485 and Ethernet solutions and expands RS-485 applications to the whole world.

The VxComm approach provides an MS-COMM-compatible interface. Therefore, previously developed programs should still function without the need for any modifications.

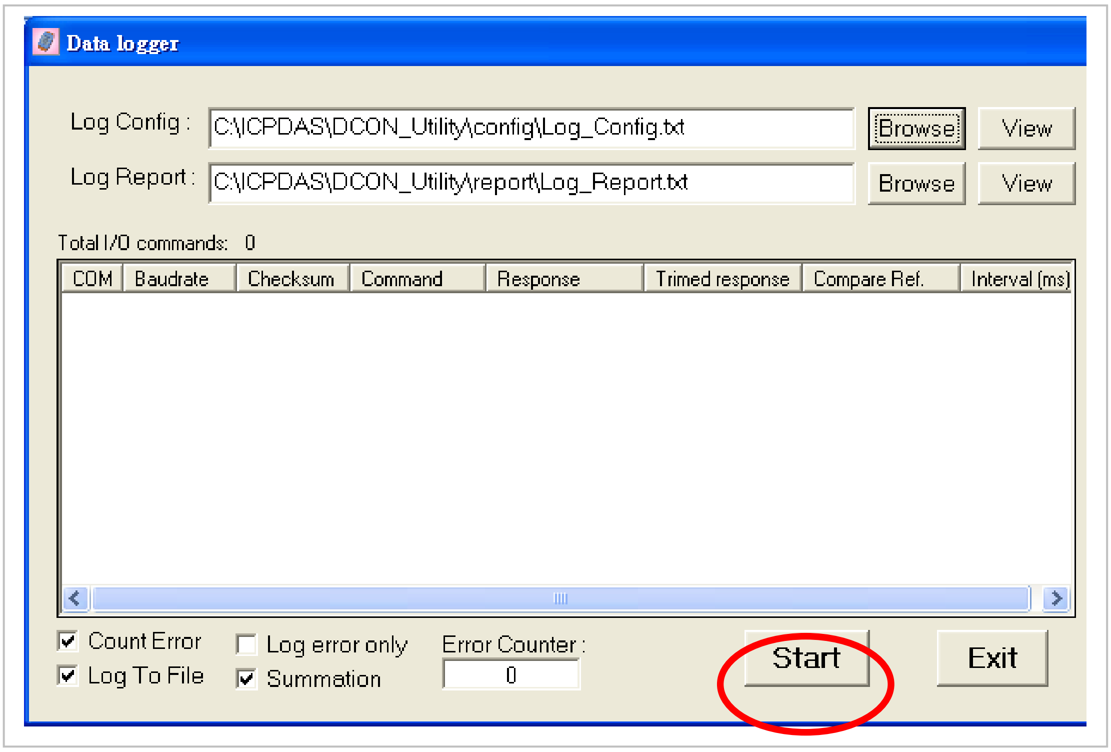
2.4 Configurable Ethernet Data Logger

Using the VxComm driver, PDS + 7000 modules can be virtualized to become COM Port + 7000 modules located on the host-PC, and then the Data Logger in the DCON Utility can be used to access data of I-7000 from the Ethernet. Signal data originating from the I-7000 modules can be analyzed using MS-Excel without the need to write any custom programs

1. The DCON utility includes a log function, as show below:



2. Configure the system connection as shown below and click the “Start” button to begin logging data.



Log Config : C:\ICPDAS\DCON_Utility\config\Log_Config.txt

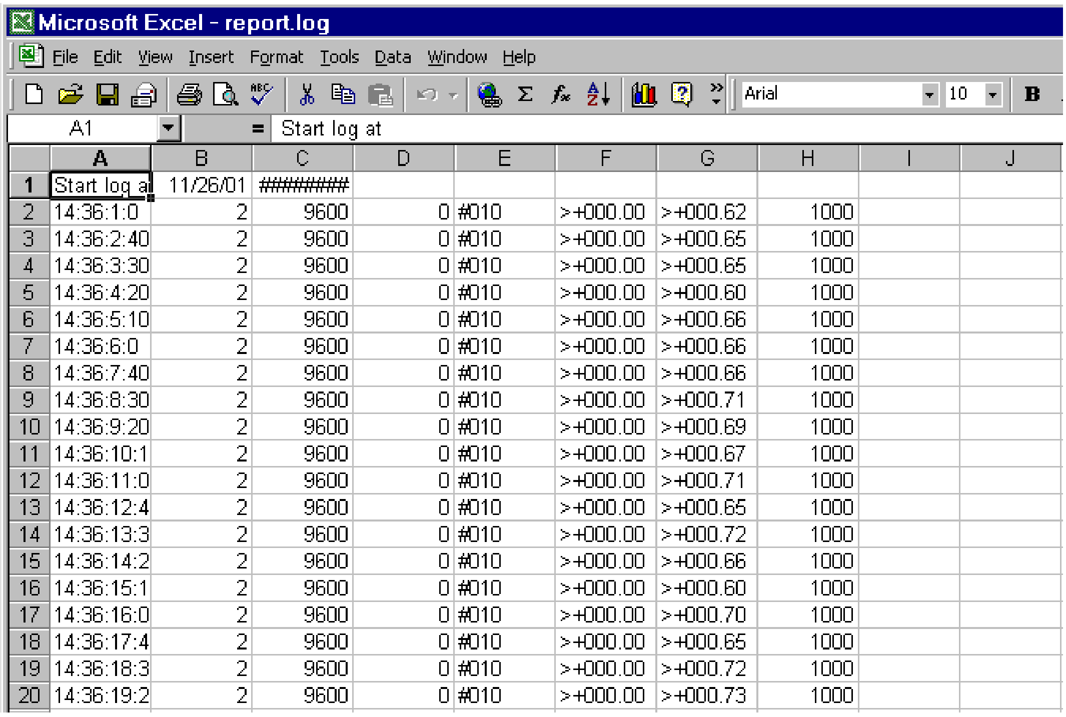
Log Report : C:\ICPDAS\DCON_Utility\report\Log_Report.txt

Total I/O commands: 0

COM	Baudrate	Checksum	Command	Response	Trimed response	Compare Ref.	Interval (ms)

☑ Count Error
☑ Log To File
☐ Log error only
☑ Summation
Error Counter : 0

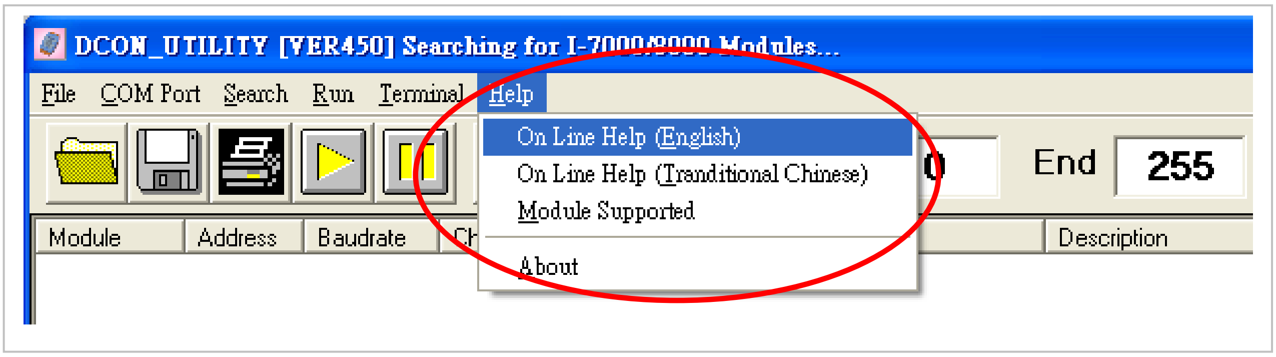
3. Open the log file in Excel to read the log data as shown in the example below:


Microsoft Excel - report.log
File Edit View Insert Format Tools Data Window Help
Arial
10
B

A1 = Start log at

	A	B	C	D	E	F	G	H	I	J
1	Start log a	11/26/01	#########							
2	14:36:1:0	2	9600	0	#010	>+000.00	>+000.62	1000		
3	14:36:2:40	2	9600	0	#010	>+000.00	>+000.65	1000		
4	14:36:3:30	2	9600	0	#010	>+000.00	>+000.65	1000		
5	14:36:4:20	2	9600	0	#010	>+000.00	>+000.60	1000		
6	14:36:5:10	2	9600	0	#010	>+000.00	>+000.66	1000		
7	14:36:6:0	2	9600	0	#010	>+000.00	>+000.66	1000		
8	14:36:7:40	2	9600	0	#010	>+000.00	>+000.66	1000		
9	14:36:8:30	2	9600	0	#010	>+000.00	>+000.71	1000		
10	14:36:9:20	2	9600	0	#010	>+000.00	>+000.69	1000		
11	14:36:10:1	2	9600	0	#010	>+000.00	>+000.67	1000		
12	14:36:11:0	2	9600	0	#010	>+000.00	>+000.71	1000		
13	14:36:12:4	2	9600	0	#010	>+000.00	>+000.65	1000		
14	14:36:13:3	2	9600	0	#010	>+000.00	>+000.72	1000		
15	14:36:14:2	2	9600	0	#010	>+000.00	>+000.66	1000		
16	14:36:15:1	2	9600	0	#010	>+000.00	>+000.60	1000		
17	14:36:16:0	2	9600	0	#010	>+000.00	>+000.70	1000		
18	14:36:17:4	2	9600	0	#010	>+000.00	>+000.65	1000		
19	14:36:18:3	2	9600	0	#010	>+000.00	>+000.72	1000		
20	14:36:19:2	2	9600	0	#010	>+000.00	>+000.73	1000		

By using the I-7000 DCON utility and MS Excel in conjunction with the **VxComm technology**, the signal data of I-7000 modules from the Ethernet network can be analyzed without the need to write custom programs. For more information about the **log function** refer to the online help feature (English and Traditional Chinese) of the DCON utility.



3. Hardware Information

3.1 Features

- Integrates Serial Devices in an Ethernet network
- Virtual COM extends the PC COM Ports
- Virtual COM supports Windows NT 4.0, 32-bit/64-bit 2000/XP/2003/Vista/7
- Watchdog Timer suitable for use in harsh environments
- Power reverse polarity protection
- Serial Port +/-4 kV ESD Protection Circuit
- Self-Tuner ASIC Controller on the RS-485 Port
- RoHS Compliant with no Halogen
- Built-in High Performance MiniOS7 from ICP DAS
- 10/100 Base-TX Ethernet, RJ-45 Port (Auto-negotiating, auto MDI/MDI-X, LED indicator)
- Low power consumption
- Palm-Sized with multiple Serial Ports
- Made from fire-retardant materials (UL94-V0 Level) (Metal for “M” versions)
- Supports D/I, Latched D/I and Counter Functions on some models
- High performance device server
- Powerful (Programmable) device server

[PPDS(M)-700(D)-MTCP only]

- Supports Modbus/TCP and Modbus/RTU
- Supports PoE (IEEE 802.3af, Class 1)

[PDS(M)-700D/PPDS(M)-700D-MTCP only]

- 5-digit LED display

[PDS-5105D-MTCP only]

- 5-digit LED display
- Serial port +/-2 kV ESD protection circuit
- 2-Port 10/100 Base-TX Ethernet Switch with LAN Bypass

3.2 Specifications

3.2.1 PDS(M)-700(D)/PPDS(M)-700(D)-MTCP

- System Specifications:

<table>
<tr><th colspan="2">Models</th><th>PDS(M)-700(D)</th><th>PPDS(M)-700(D)-MTCP</th></tr>
<tr><td colspan="4"><b>CPU</b></td></tr>
<tr><td colspan="2">CPU</td><td colspan="2">80186-80 MHz or compatible</td></tr>
<tr><td colspan="2">SRAM</td><td colspan="2">512 KB</td></tr>
<tr><td colspan="2">Flash Memory</td><td colspan="2">Flash ROM: 512 KB; Erase unit is one sector (64 KB);
1000,000 erase/write cycles</td></tr>
<tr><td colspan="2">EEPROM</td><td colspan="2">16 KB; Data retention:40 years; 1000,000 erase/write cycles</td></tr>
<tr><td colspan="2">Built-in Watchdog Timer</td><td colspan="2">Yes</td></tr>
<tr><td colspan="4"><b>Communication Interface</b></td></tr>
<tr><td rowspan="2">Non-isolated</td><td>COM1</td><td colspan="2">RS-232 (TxD, RxD, RTS, CTS, GND)</td></tr>
<tr><td>COM2</td><td colspan="2">RS-485 (D2+, D2-, GND)</td></tr>
<tr><td colspan="2">Ethernet</td><td colspan="2">10/100 Base-Tx, RJ-45 port
(Auto-negotiating, auto MDI/MDI-X, LED indicator)</td></tr>
<tr><td colspan="2">PoE</td><td>-</td><td>IEEE 802.3 af</td></tr>
<tr><td colspan="4"><b>COM Port Formats</b></td></tr>
<tr><td colspan="2">Data Bit</td><td colspan="2">7,8: for COM1, COM2
5,6,7,8: for COM3 ~ COM4</td></tr>
<tr><td colspan="2">Parity</td><td colspan="2">None, Even, Odd, Mark, Space</td></tr>
<tr><td colspan="2">Stop Bit</td><td colspan="2">1: for COM, COM2
1,2: for COM3 ~ COM4</td></tr>
<tr><td colspan="2">Baud Rate</td><td colspan="2">115200 bps Max.</td></tr>
<tr><td colspan="4"><b>LED Indicators</b></td></tr>
<tr><td colspan="2">5-digit 7 Segment</td><td colspan="2">Yes (Display for "D") versions</td></tr>
<tr><td colspan="2">System</td><td colspan="2">Red</td></tr>
<tr><td colspan="2">PoE</td><td>-</td><td>Green</td></tr>
<tr><td colspan="4"><b>Power</b></td></tr>
<tr><td colspan="2">Protection</td><td colspan="2">Power Reverse Polarity Protection</td></tr>
<tr><td colspan="2">Required Supply Voltage</td><td>+10 V_{DC} ~ +30 V_{DC} (non-regulated)</td><td>+12 V_{DC} ~+48 V_{DC} (non-regulated)
PoE (IEEE 802.3af, Class 1)</td></tr>
<tr><td colspan="2">Power Consumption</td><td>PDS(M)-700: 2.0 W
PDS(M)-700D: 2.7 W</td><td>PPDS(M)-700-MTCP: 2.2 W
PPDS(M)-700D-MTCP: 2.9 W</td></tr>
</table>

- I/O Specifications:

Models		PDS(M)-700(D)	PPDS(M)-700(D)-MTCP
Digital Output			
Output Type		Open Collector (Sink/NPN)	
Load Voltage		30 V_{DC} max.	
Load Current		100 mA max.	
Isolated Voltage		Non-isolated	
Digital Input			
Input Type		Source (Dry Type), Common Ground	
Off Voltage Level		+1 V max.	
On Voltage Level		+3.5 V ~ +30 V	
Isolated Voltage		Non-isolated	
Counter	Max. Count	16-bit (65535)	
	Max. Input Frequency	100 Hz	
	Min. Pulse Width	5 ms	

3.2.2 DS-700/PPDS-700-MTCP

Models	DS-700	PPDS-700-MTCP
CPU		
CPU	80186-80 MHz or compatible	
SRAM	512 KB	
Flash Memory	Flash ROM: 512 KB; Erase unit is one sector (64 KB); 1000,000 erase/write cycles	
EEPROM	16 KB; Data retention: 40 years; 1000,000 erase/write cycles	
Built-in Watchdog Timer	Yes	
COM Port Formats		
Data Bit	7,8	
Parity	None, Even, Odd, Mark, Space	
Stop Bit	1	
Baud Rate	115200 bps Max.	
LED Indicators		
L1	Run (Red)	
L2	Link/Act (Red)	
L3	10/100 M (Orange)	
PoE	-	Green
Power		
Protection	Power Reverse Polarity Protection	
Required Supply Voltage	+12 V_{DC} ~ +48 V_{DC} (non-regulated)	+12 V_{DC} ~ +48 V_{DC} (non-regulated) PoE (IEEE802.3af, Class 1)
Power Consumption	2.0 W	2.2 W

Communication Interface			
Models		**DS-712**	**PPDS-712-MTCP**
Non-isolated	COM1	RS-232 (TxD, RxD, RTS, CTS, GND)	
Ethernet		10/100 Base-TX, RJ-45 Port (Auto-negotiating, auto MDI/MDI-X, LED indicator)	
PoE		-	IEEE 802.3af
Models		**DS-715**	**PPDS-715-MTCP**
Isolated (2000 Vrms)	COM1	RS-422 (TxD+, TxD-, RxD+, RxD-) RS-485 (D2+, D2-)	
RS-485	Bias Resistor	Yes, 1 KΩ	
	Node	245 (max.)	
Ethernet		10/100 Base-Tx, RJ-45 port (Auto-negotiating, auto MDI/MDI-X, LED indicator)	
PoE		-	IEEE 802.3af

3.2.3 PPDS-700-IP67

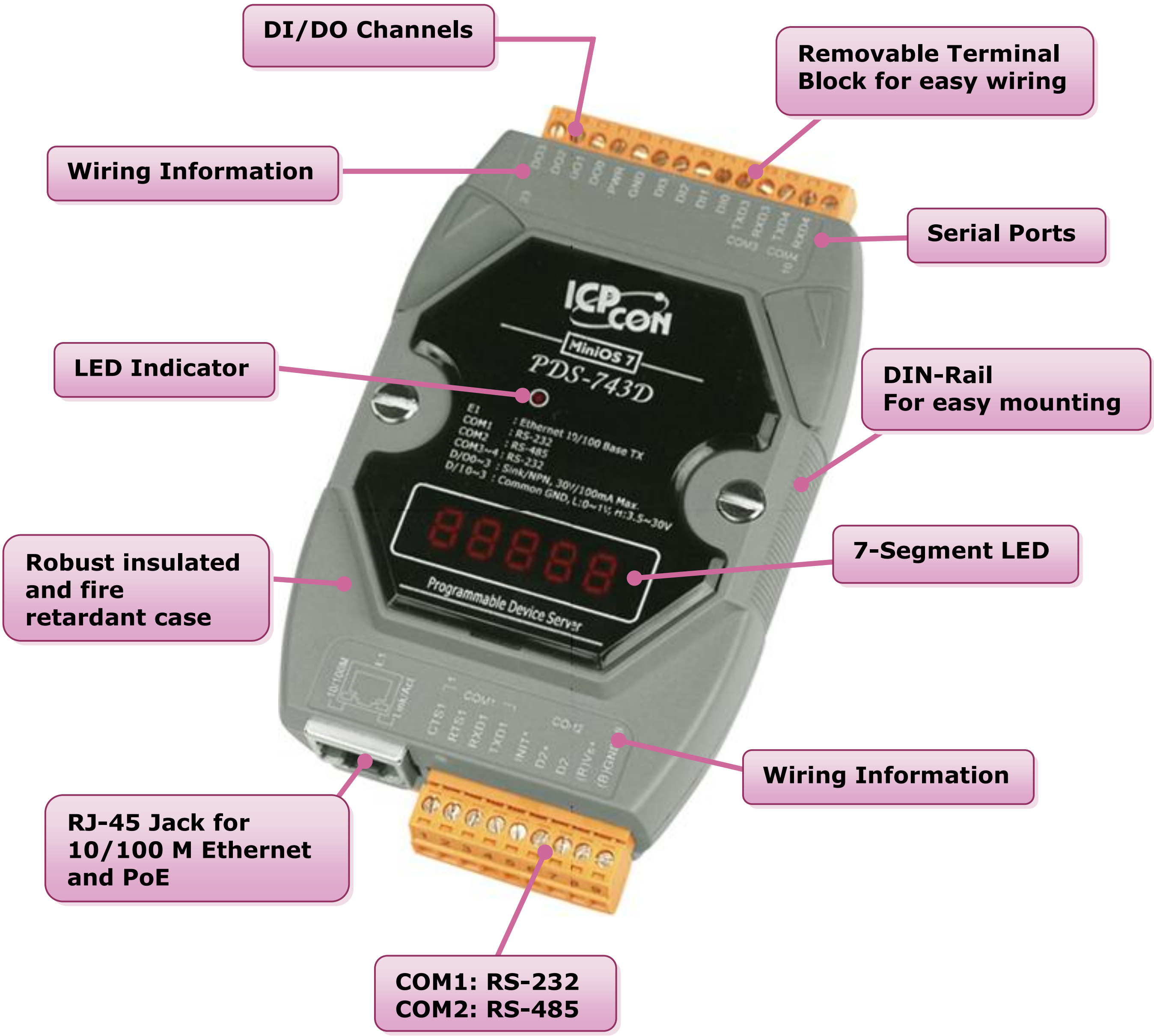
Models		PPDS-700-IP67	
CPU			
CPU		80186-80 MHz or compatible	
SRAM		512 KB	
Flash Memory		Flash ROM: 512 KB; Erase unit is one sector (64 KB); 1000,000 erase/write cycles	
EEPROM		16 KB; Data retention:40 years; 1000,000 erase/write cycles	
Built-in Watchdog Timer		Yes	
Communication Interface			
Non-isolated	COM1	RS-232 (TxD, RxD, RTS, CTS, GND)	
	COM2	RS-485 (D2+, D2-, GND)	
RS-485	Bias Resistor	Yes, 1 KΩ	
	Node	245 (max.)	
Ethernet		10/100 Base-Tx, RJ-45 port (Auto-negotiating, auto MDI/MDI-X, LED indicator)	
PoE		-	IEEE 802.3 af
COM Port Formats			
Data Bit		7,8: for COM1, COM2 5,6,7,8: for COM3 ~ COM4	
Parity		None, Even, Odd, Mark, Space	
Stop Bit		1: for COM, COM2 1,2: for COM3 ~ COM4	
Baud Rate		115200 bps Max.	
LED Indicators			
Ethernet		Green: Link/Act (E1) Orange: 10/100 M (E1)	
System		Red: Sys	
COM1~COM4		Green: RxD Orange: TxD	
Power			
Protection		Power Reverse Polarity Protection	
Required Supply Voltage		+12 V_{DC} ~+48 V_{DC} (non-regulated) PoE (IEEE 802.3af, Class 1)	
Power Consumption		2.2 W	

3.2.4 PDS-5105D-MTCP

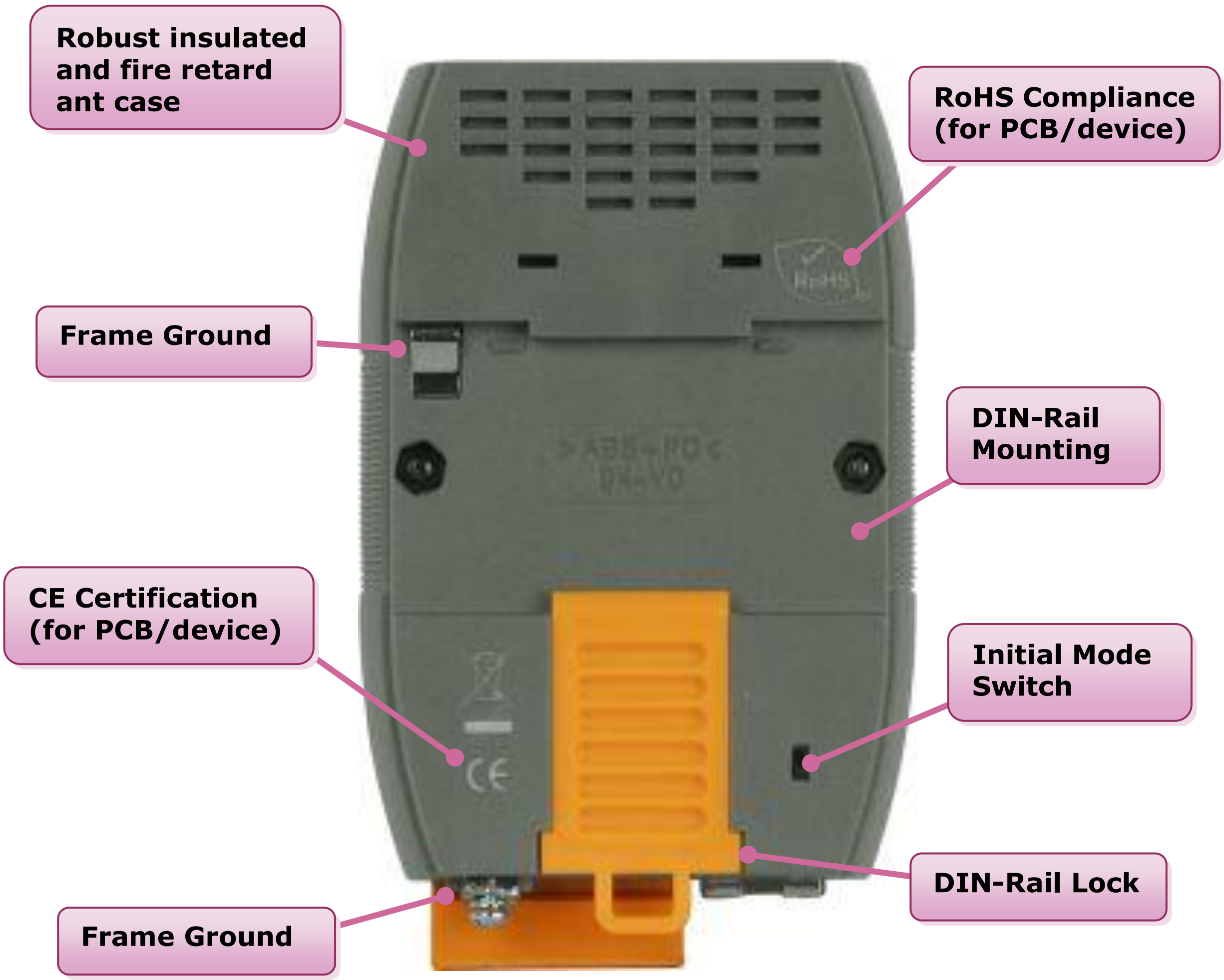
Models		PDS-5105D-MTCP
CPU		
CPU		80186 or compatible (16-bit and 80 MHz)
SRAM		512 KB
Flash Memory		512 KB; Erase unit is one sector (64 KB); 1000,000 erase/write cycles
EEPROM		16 KB
Watchdog Timer		Yes (0.8 seconds)
Communication Ports		
Non-isolated	COM1	RS-232 (TxD, RxD, GND)/RS-485(D1+,D1-), Self-tuner ASIC inside
	COM2~10	RS-485 (Dx+, Dx-), Self-tuner ASIC inside
RS-485	Bias Resistor	Yes, 1 KΩ
	Node	254 (max.)
Ethernet		2-port 10/100 Base-Tx Ethernet Switch with LAN Bypass, RJ-45 x2 (Auto-negotiating, auto MDI/MDI-X, LED indicator)
COM Port Formats		
Data Bit		7,8: for COM1, COM2 5,6,7,8: for COM3 ~ COM10
Parity		None, Even, Odd, Mark, Space
Stop Bit		1,2: for COM1 ~ COM10
Baud Rate		115200 bps Max. @ 10 Ports, half-duplex, 80% loading
LED Indicators		
5-digit 7 Segment		Yes
System		Red
Mechanical		
Dimensions (W x H x D)		91 mm x 132 mm x 52 mm
Installation		DIN-Rail
Case		Fire-Retardant Plastic (UL94-V0 Level)
Environmental		
Operating Temperature		-25° ~ 75°C
Storage Temperature		-30° ~ 85°C
Humidity		10 ~ 90% RH, non-condensing
Power Consumption		0.05 A @ 24 V_{DC}
Power		
Protection		Power reverse polarity protection
Frame GND		Yes (for EMS Protection)
Input Range		+12 ~+48 V_{DC} (non-regulated)
Power Consumption		4.8 W

3.3 Front View

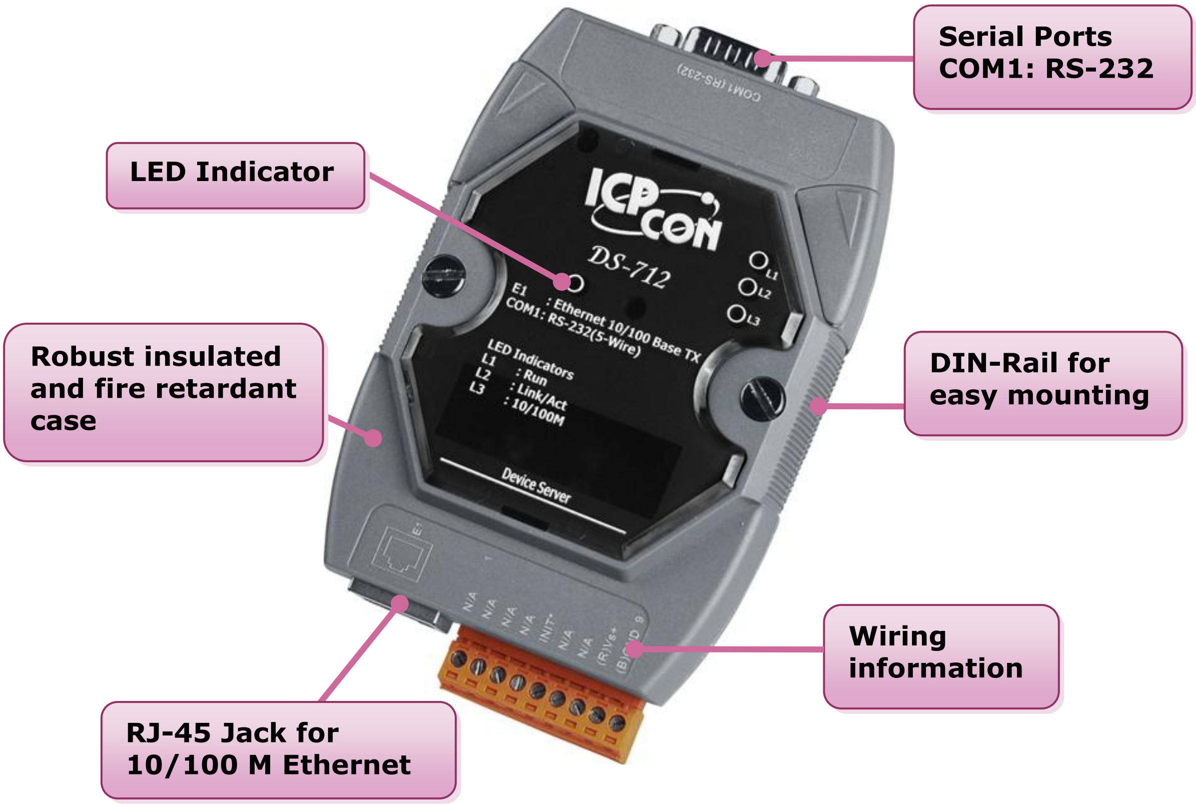
3.3.1 PDS-700/PPDS-700-MTCP Front View

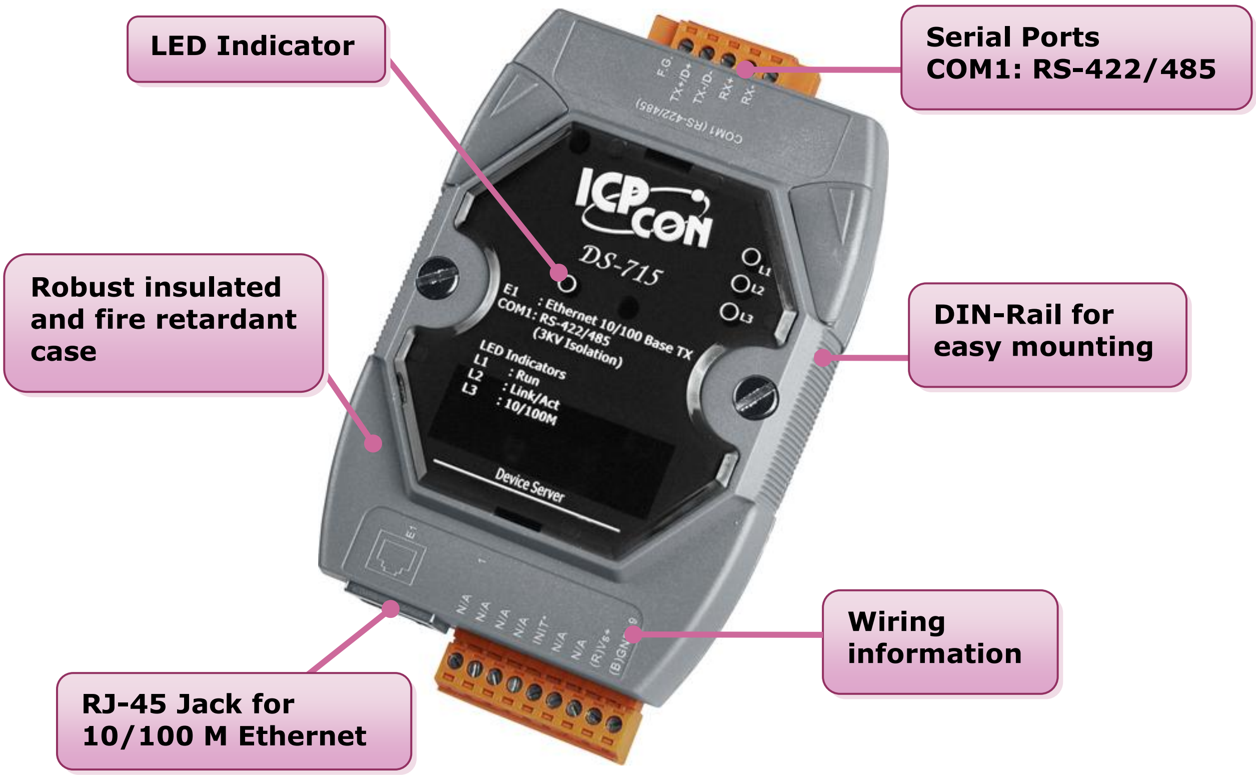

DI/DO Channels
Removable Terminal Block for easy wiring
Wiring Information
Serial Ports
LED Indicator
DIN-Rail For easy mounting
7-Segment LED
Robust insulated and fire retardant case
Wiring Information
RJ-45 Jack for 10/100 M Ethernet and PoE
COM1: RS-232
COM2: RS-485
ICP CON
MiniOS 7
PDS-743D
Programmable Device Server

3.3.2 PDS-700/PPDS-700-MTCP Rear View

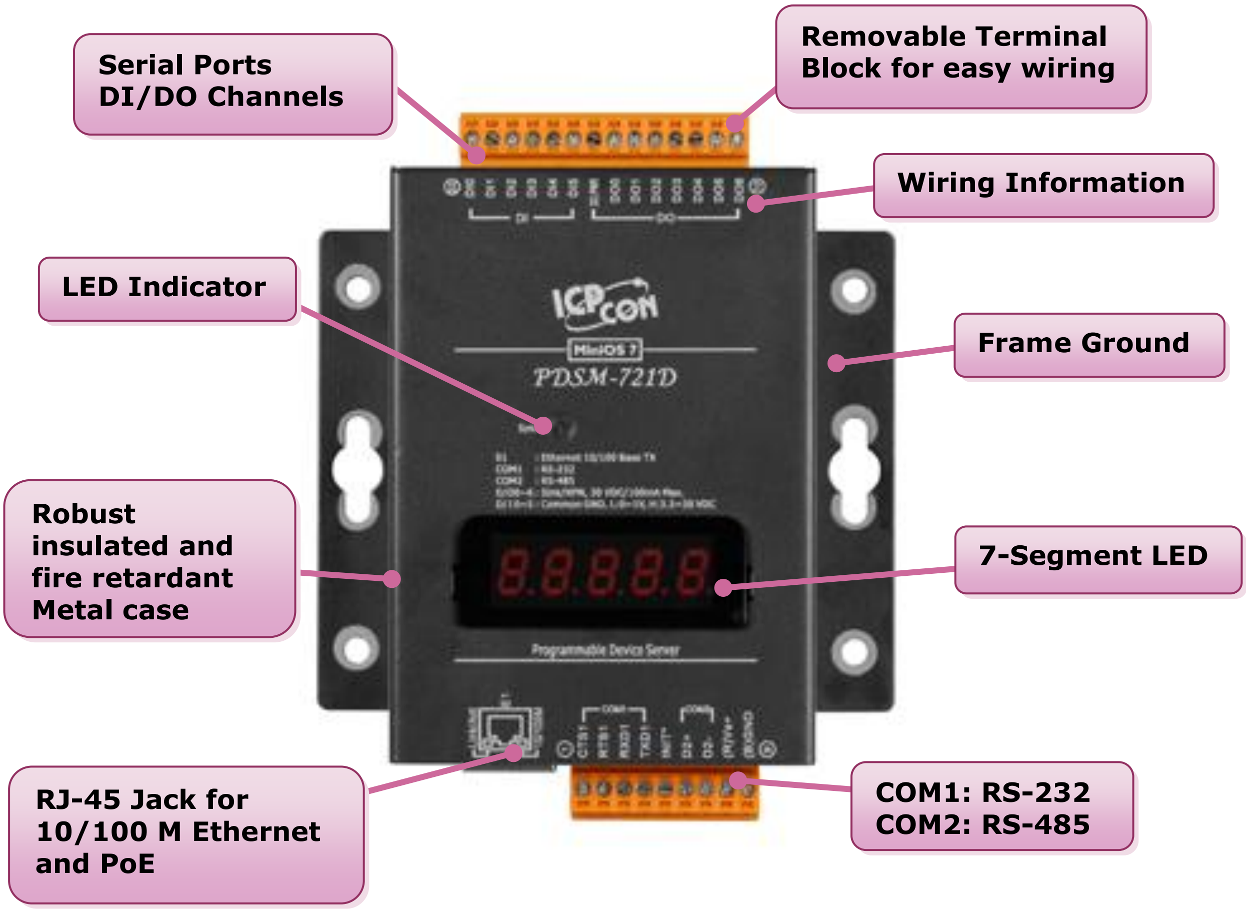

Robust insulated and fire retard ant case
RoHS Compliance (for PCB/device)
Frame Ground
DIN-Rail Mounting
CE Certification (for PCB/device)
Initial Mode Switch
DIN-Rail Lock
Frame Ground

3.3.3 DS-700 Front View

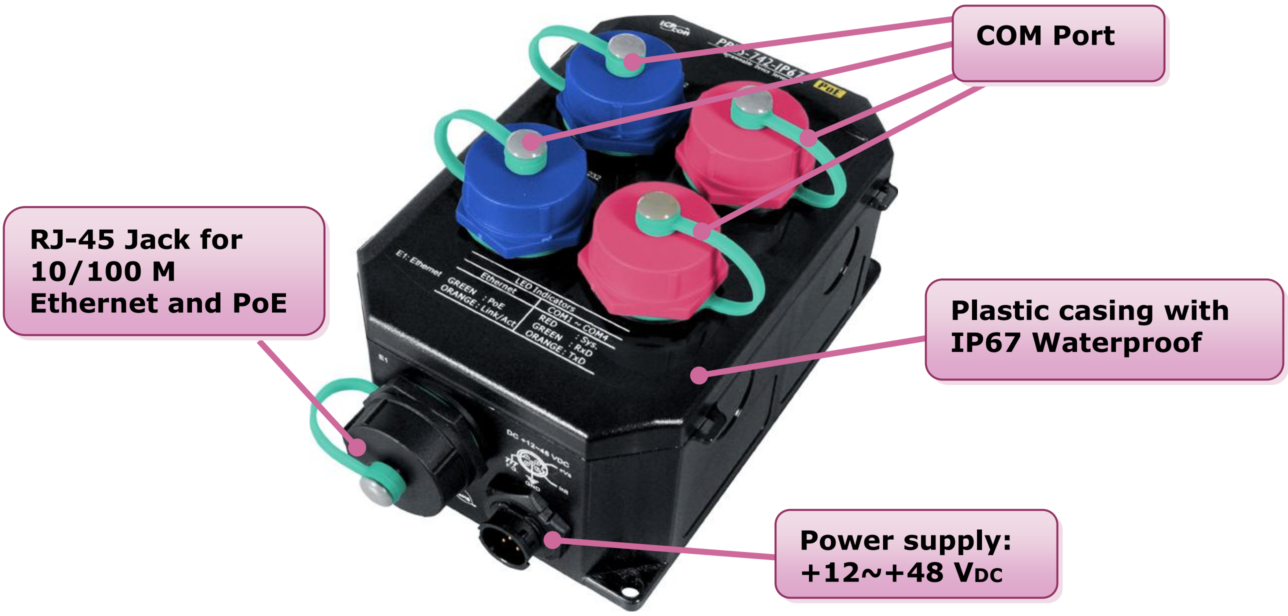




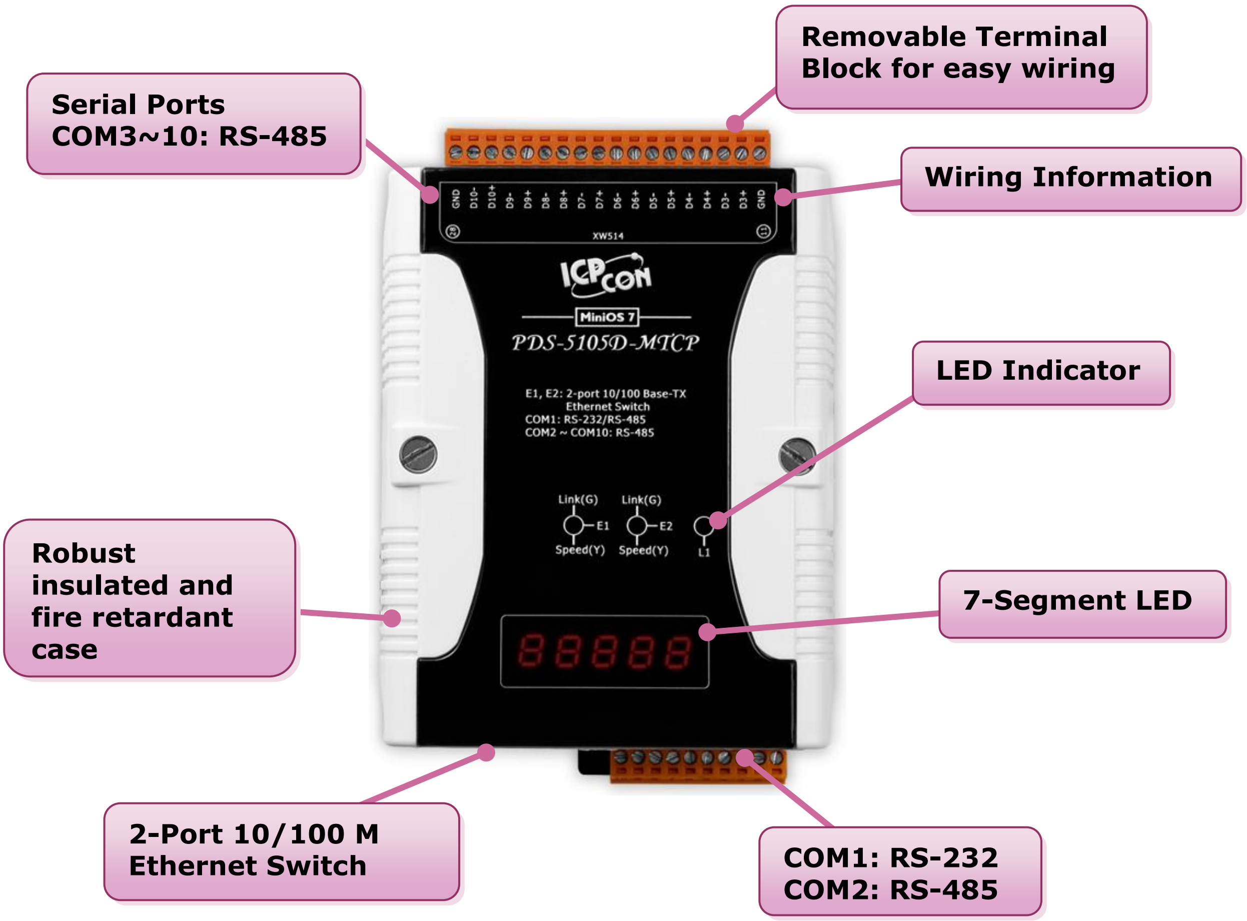
3.3.4 PDSM-700/PPDSM-700-MTCP Front View


Serial Ports
DI/DO Channels
Removable Terminal Block for easy wiring
Wiring Information
LED Indicator
Frame Ground
Robust insulated and fire retardant Metal case
7-Segment LED
RJ-45 Jack for 10/100 M Ethernet and PoE
COM1: RS-232
COM2: RS-485
PDSM-721D
Programmable Device Server

3.3.5 PPDS-700-IP67 Front View


COM Port
RJ-45 Jack for 10/100 M Ethernet and PoE
Plastic casing with IP67 Waterproof
Power supply: +12~+48 VDC

3.3.6 PDS-5105D-MTCP Front View


Removable Terminal Block for easy wiring
Serial Ports COM3~10: RS-485
Wiring Information
LED Indicator
Robust insulated and fire retardant case
7-Segment LED
2-Port 10/100 M Ethernet Switch
COM1: RS-232 COM2: RS-485
ICPCON
MiniOS 7
PDS-5105D-MTCP
E1, E2: 2-port 10/100 Base-TX Ethernet Switch
COM1: RS-232/RS-485
COM2 ~ COM10: RS-485
Link(G)
Link(G)
E1
E2
Speed(Y)
Speed(Y)
L1

3.4 PDS Series Comparison Table

- PDS(M)-700(D)/PPDS(M)-700(D)-MTCP Series Programmable Device Servers

Models						RS-232 RS-485 RS-422/485	DI/DO
	PDS	**M**	**-7 □ □**	**D**			
P	**PDS**	**M**	**-7 □ □**	**D**	**-MTCP**		
PoE	Programmable Device Server	Metal		LED Display	Modbus/TCP		
PDS -720 D PPDS -720 D -MTCP						1 RS-232 1 RS-485	-
PDS M-721 D PPDS M-721 D -MTCP						1 RS-232 1 RS-485	6/7
PDS M-732 D PPDS M-732 D -MTCP						2 RS-232 1 RS-485	4/4
PDS M-734 D PPDS M-734 D -MTCP						1 RS-232 1 RS-485 1 RS-422/485	4/4
PDS M-742 D PPDS M-742 D -MTCP						3 RS-232 1 RS-485	-
PDS M-743 D PPDS M-743 D -MTCP						3 RS-232 1 RS-485	4/4
PDS M-752 D PPDS M-752 D -MTCP						4 RS-232 1 RS-485	-
PDS M-755 D PPDS M-755 D -MTCP						1 RS-232 4 RS-485	-
PDS M-762 D PPDS M-762 D -MTCP						5 RS-232 1 RS-485	1/2
PDS M-782 D PPDS M-782 D -MTCP						7 RS-232 1 RS-485	-
PPDS M-782 D -25/D6						7 RS-232 1 RS-485	-

Note:

1. PPDS(M)-700(D)-MTCP supports PoE and Modbus Gateway.
2. **D** versions support 7-segment 5-digit LED display.
3. **M** versions is equipped with metal case.

DS-700 Series Non-Programmable Device Servers

Models					RS-232 RS-485 RS-422/485	DI/DO
		DS	-7 □ □			
P	P	DS	-7 □ □	-MTCP		
PoE	Programmable	Device Server		Modbus/TCP		
DS -712 PPDS -712 -MTCP					1 RS-232	-
DS -715 PPDS -715 -MTCP					1 RS-422/485	-
Note: 1. PPDS-700-MTCP supports PoE and Modbus Gateway.						

PPDS-700-IP67 Series Programmable Device Servers

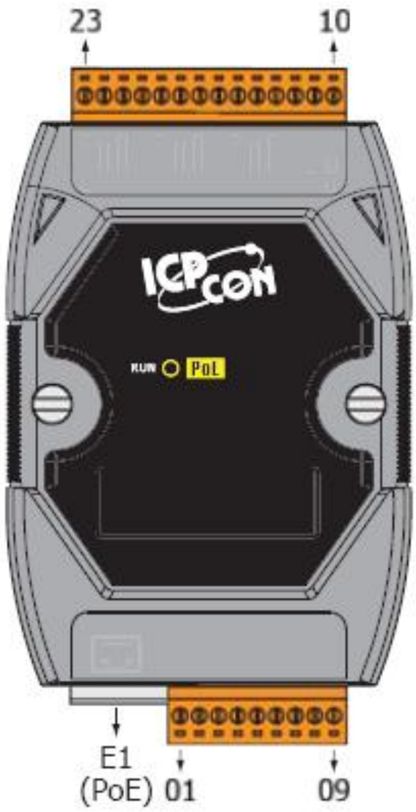
Models				RS-232 RS-485 RS-422/485	DI/DO
P	PDS	-7 □ □	-IP67		
PoE	Programmable Device Server		Waterproof Protection		
PPDS -741-IP67				1 RS-232 3 RS-485	-
PPDS -742-IP67				2 RS-232 2 RS-485	-
PPDS -743-IP67				3 RS-232 1 RS-485	-
Note: 1. **IP67** version is equipped with IP67 case.					

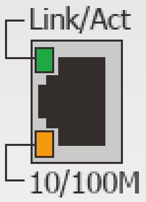
PDS-5105D-MTCP Programmable Device Servers

Models				RS-232 RS-485 RS-422/485	DI/DO
PDS	-5 □ □ □	D	-MTCP		
Programmable Device Server		LED Display	Modbus/TCP		
PPDS -5105D -MTCP				1 RS-232 10 RS-485	-
Note: 1. PPDS-5000(D)-MTCP supports PoE, 2-port Switch and Modbus Gateway. 2. **D** versions support 7-segment 5-digit LED display.					

3.5 Pin Assignments

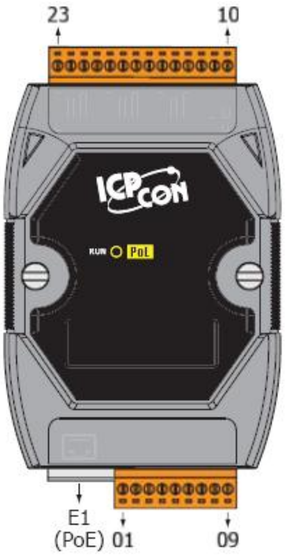
3.5.1 PDS-720(D)/PPDS-720(D)-MTCP

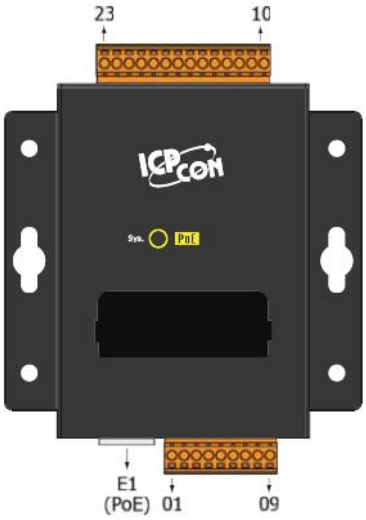


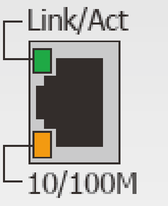
Terminal No.		Pin Assignment
E1		
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)

3.5.2 PDS(M)-721(D)/ PPDS(M)-721(D)-MTCP



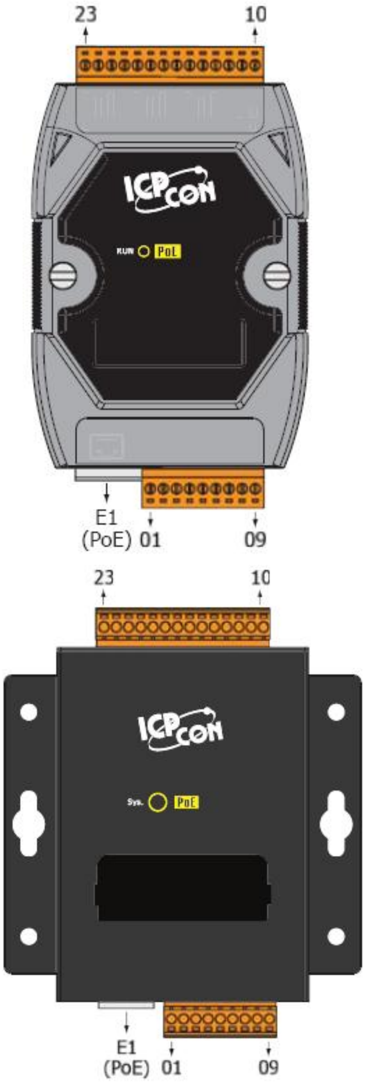


Terminal No.		Pin Assignment
E1		
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

Terminal No.		Pin Assignment
DI	23	DI0
	22	DI1
	21	DI2
	20	DI3
	19	DI4
	18	DI5
DO	17	DO.PWR
	16	DO0
	15	DO1
	14	DO2
	13	DO3
	12	DO4
	11	DO5
	10	DO6

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS(M)-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS(M)-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)
DO.PWR	Power Input for Digital Output.

3.5.3 PDS(M)-732(D)/ PPDS(M)-732(D)-MTCP

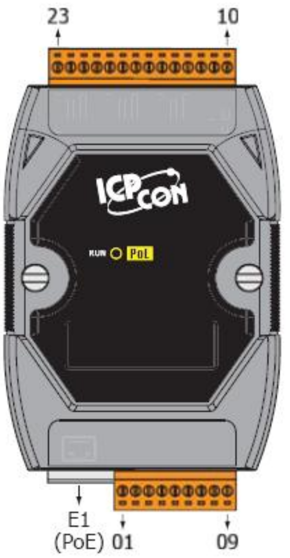


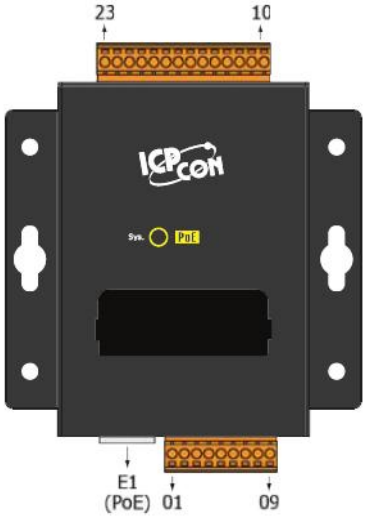
Terminal No.		Pin Assignment
E1		Link/Act 10/100M
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

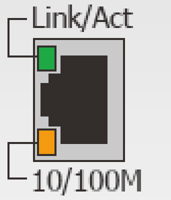
Terminal No.		Pin Assignment
DO	23	DO3
	22	DO2
	21	DO1
	20	DO0
	19	DO.PWR
	18	GND
DI	17	DI3
	16	DI2
	15	DI1
	14	DI0
COM3	13	RxD3
	12	TxD3
	11	RTS3
	10	CTS3

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS(M)-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS(M)-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)
DO.PWR	Power Input for Digital Output.

3.5.4 PDS(M)-734(D)/ PPDS(M)-734(D)-MTCP



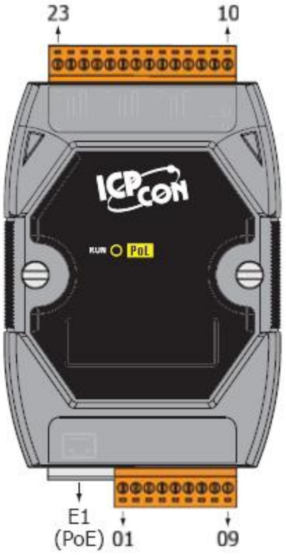


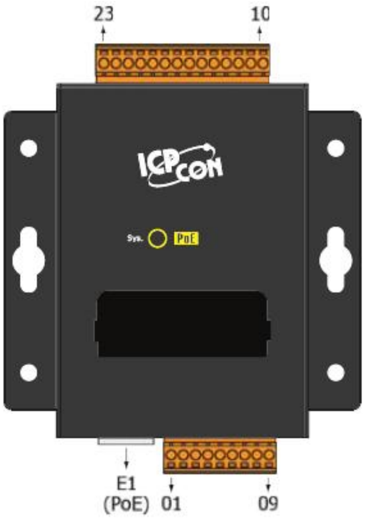
Terminal No.		Pin Assignment
E1		 Link/Act 10/100M
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

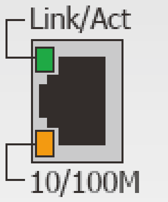
Terminal No.		Pin Assignment
DO	23	DO3
	22	DO2
	21	DO1
	20	DO0
	19	DO.PWR
	18	GND
DI	17	DI3
	16	DI2
	15	DI1
	14	DI0
COM3	13	RxD3-
	12	RxD3+
	11	TxD3-/D3-
	10	TxD3+/D3+

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS(M)-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS(M)-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)
DO.PWR	Power Input for Digital Output.

3.5.5 PDS(M)-742(D)/PPDS(M)-742(D)-MTCP



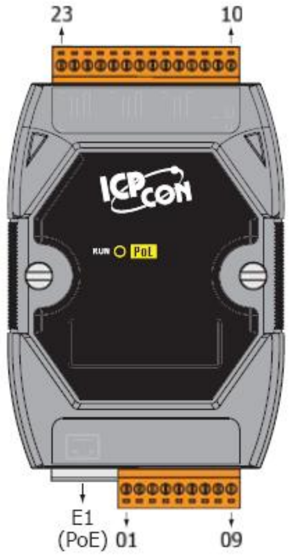


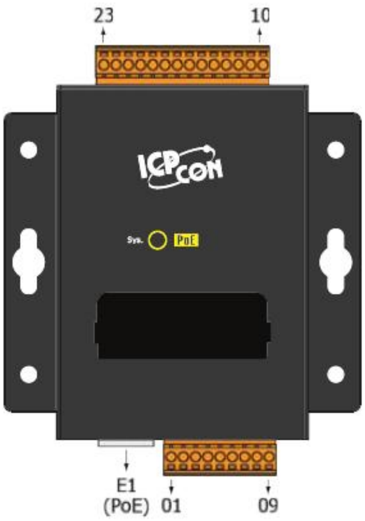
Terminal No.		Pin Assignment
E1		 Link/Act 10/100M
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

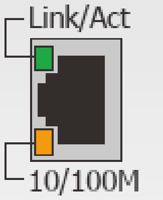
Terminal No.		Pin Assignment
COM4	23	DI4
	22	DCD4
	21	DTR4
	20	DSR4
	19	CTS4
	18	RTS4
	17	TxD4
	16	RxD4
	15	GND4
COM3	14	GND3
	13	RxD3
	12	TxD3
	11	RTS3
	10	CTS3

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS(M)-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS(M)-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)

3.5.6 PDS(M)-743(D)/PPDS(M)-743(D)-MTCP



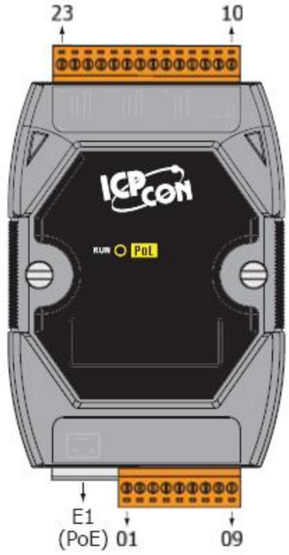


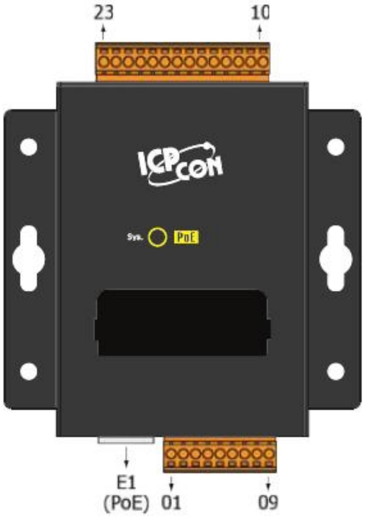
Terminal No.		Pin Assignment
E1		 Link/Act 10/100M
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

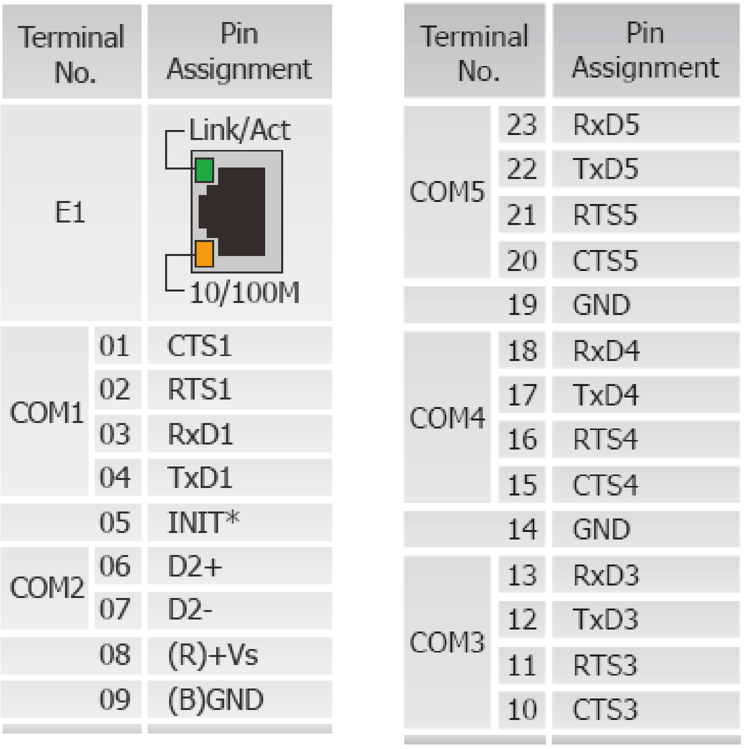
Terminal No.		Pin Assignment
DO	23	DO3
	22	DO2
	21	DO1
	20	DO0
	19	DO.PWR
	18	GND
DI	17	DI3
	16	DI2
	15	DI1
	14	DI0
COM3	13	TxD3
	12	RxD3
COM4	11	TxD4
	10	RxD4

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS(M)-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS(M)-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)
DO.PWR	Power Input for Digital Output.

3.5.7 PDS(M)-752(D)/PPDS(M)-752(D)-MTCP





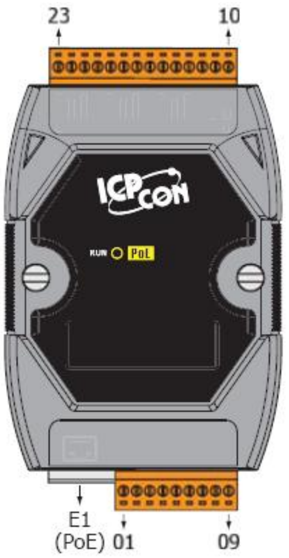


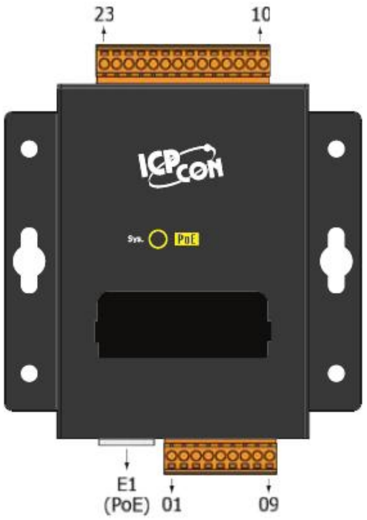
Terminal No.		Pin Assignment
E1		Link/Act 10/100M
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

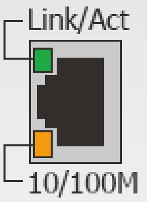
Terminal No.		Pin Assignment
COM5	23	RxD5
	22	TxD5
	21	RTS5
	20	CTS5
	19	GND
COM4	18	RxD4
	17	TxD4
	16	RTS4
	15	CTS4
	14	GND
COM3	13	RxD3
	12	TxD3
	11	RTS3
	10	CTS3

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS(M)-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS(M)-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)

3.5.8 PDS(M)-755(D)/PPDS(M)-755(D)-MTCP



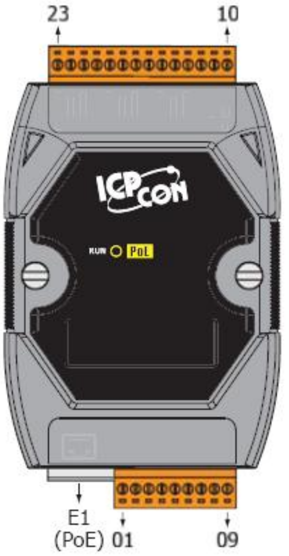


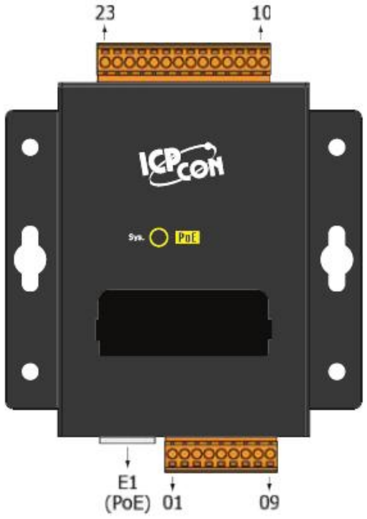
Terminal No.		Pin Assignment
E1		
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

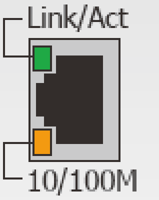
Terminal No.		Pin Assignment
COM5	23	DATA+
	22	DATA-
	21	--
	20	--
	19	--
	18	--
COM4	17	DATA+
	16	DATA-
	15	--
	14	--
	13	--
	12	--
COM3	11	DATA+
	10	DATA-

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS(M)-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS(M)-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)

3.5.9 PDS(M)-762(D)/PPDS(M)-762(D)-MTCP



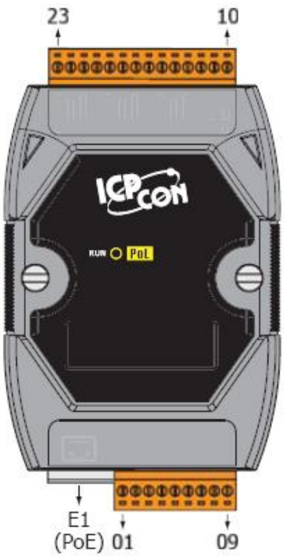


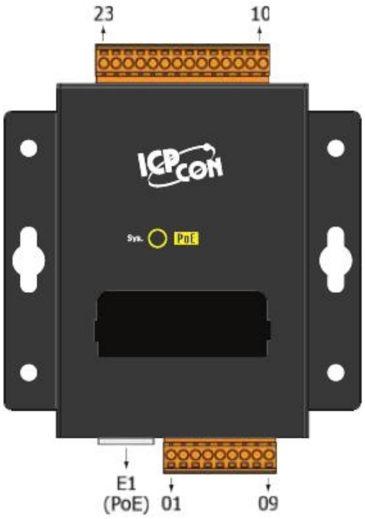
Terminal No.		Pin Assignment
E1		
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

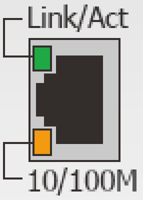
Terminal No.		Pin Assignment
DO	23	DO0
	22	DO1
	21	DO.PWR
DI	20	DI0
	19	GND
COM6	18	TxD6
	17	RxD6
COM5	16	TxD5
	15	RxD5
	14	GND
COM4	13	TxD4
	12	RxD4
COM3	11	TxD3
	10	RxD3

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS(M)-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS(M)-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)
DO.PWR	Power Input for Digital Output.

3.5.10 PDS(M)-782(D)/PPDS(M)-782(D)-MTCP



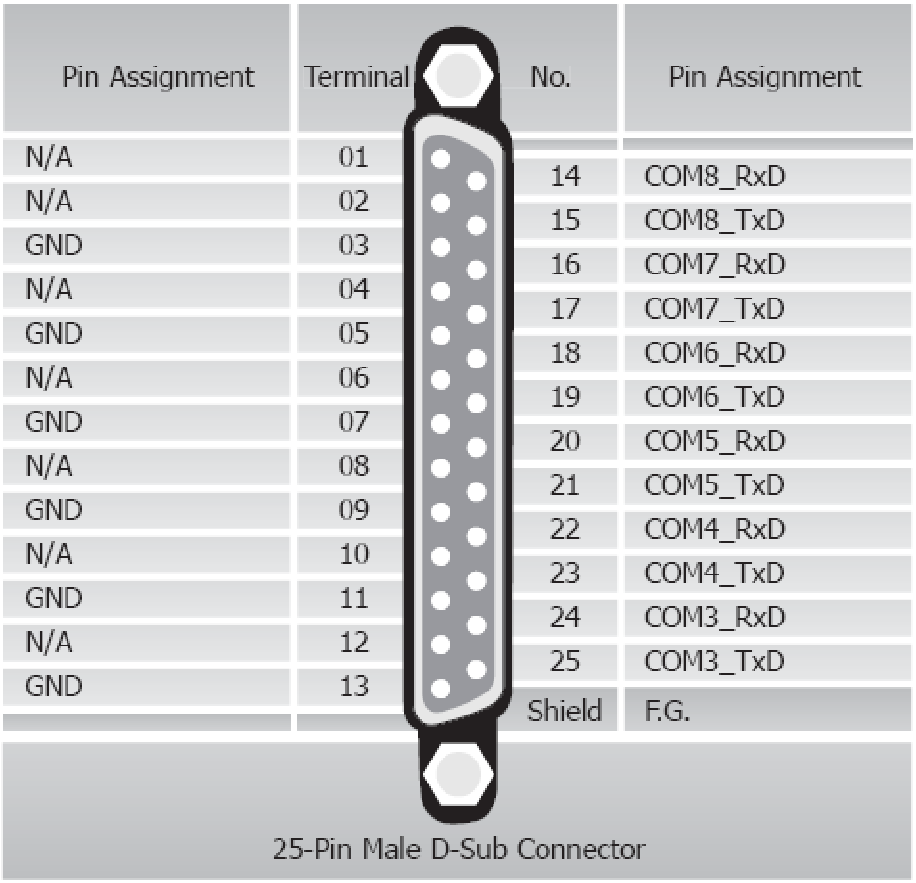


Terminal No.		Pin Assignment
E1		 Link/Act 10/100M
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	INIT*
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

Terminal No.		Pin Assignment
COM8	23	TxD8
	22	RxD8
COM7	21	TxD7
	20	RxD7
	19	GND
COM6	18	TxD6
	17	RxD6
COM5	16	TxD5
	15	RxD5
	14	GND
COM4	13	TxD4
	12	RxD4
COM3	11	TxD3
	10	RxD3

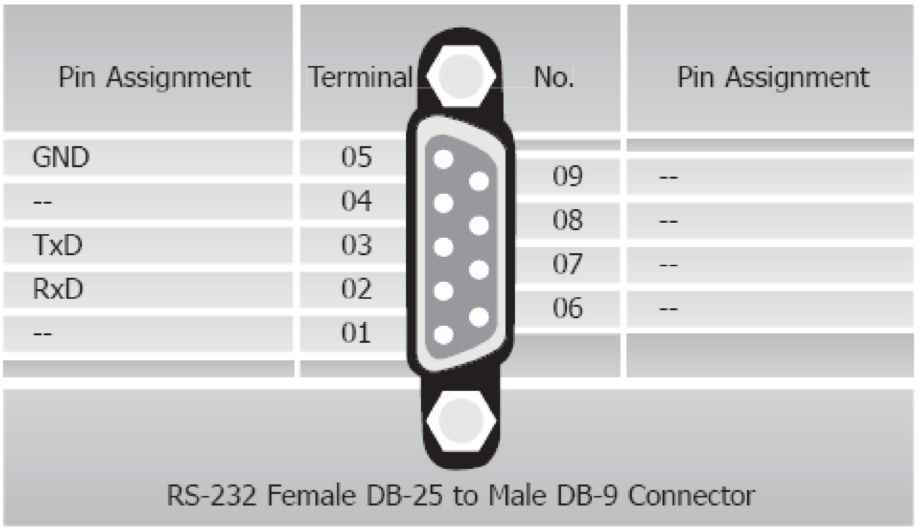
INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply. PDS(M)-700(D) series: +10 ~ +30 V_{DC} unregulated; PPDS(M)-700(D)-MTCP series: +12 ~ +48 V_{DC} unregulated
(B)GND	GND Pin for the power supply (COM1 GND)

3.5.11 PDS-782(D)-25/D6



Pin Assignment	Terminal No.		Pin Assignment
N/A	01	14	COM8_RxD
N/A	02	15	COM8_TxD
GND	03	16	COM7_RxD
N/A	04	17	COM7_TxD
GND	05	18	COM6_RxD
N/A	06	19	COM6_TxD
GND	07	20	COM5_RxD
N/A	08	21	COM5_TxD
GND	09	22	COM4_RxD
N/A	10	23	COM4_TxD
GND	11	24	COM3_RxD
N/A	12	25	COM3_TxD
GND	13	Shield	F.G.

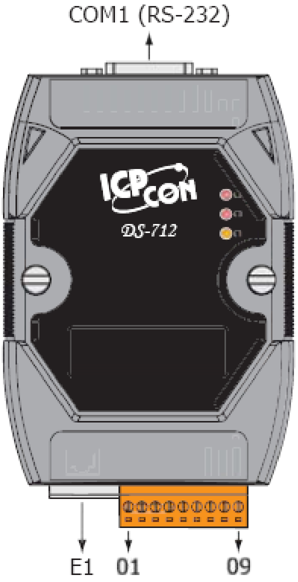
25-Pin Male D-Sub Connector

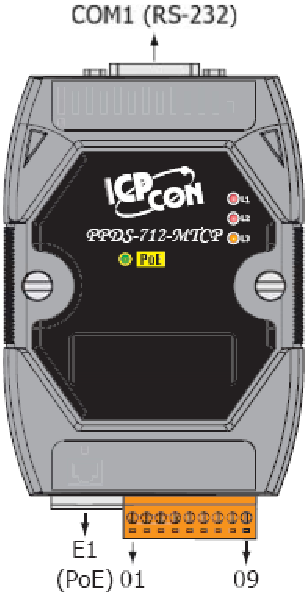


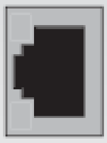
Pin Assignment	Terminal No.		Pin Assignment
GND	05	09	--
--	04	08	--
TxD	03	07	--
RxD	02	06	--
--	01		

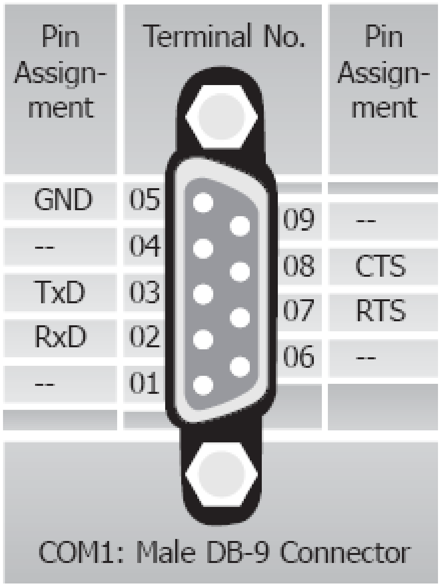
RS-232 Female DB-25 to Male DB-9 Connector

3.5.12 DS-712/PPDS-712-MTCP





Terminal No.	Pin Assignment
E1	
01	N/A
02	N/A
03	N/A
04	N/A
05	INIT*
06	N/A
07	N/A
08	(R)+Vs
09	(B)GND

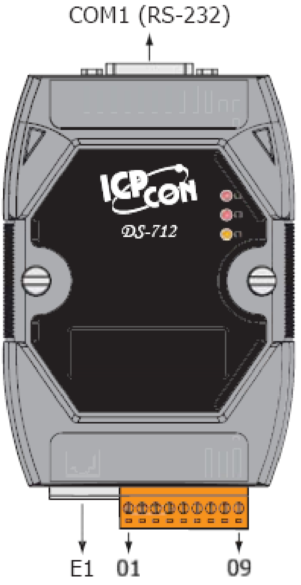


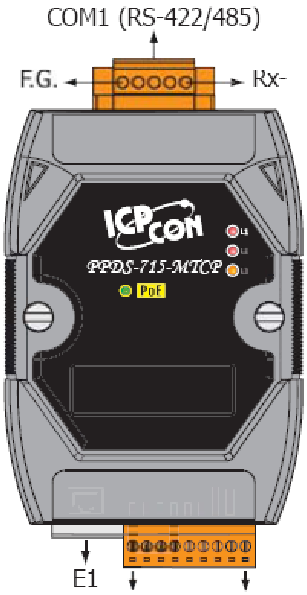
Pin Assign-ment	Terminal No.		Pin Assign-ment
GND	05	09	--
--	04	08	CTS
TxD	03	07	RTS
RxD	02	06	--
--	01		

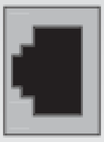
COM1: Male DB-9 Connector

INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply (+12 ~ +48 V_{DC} unregulated)
(B)GND	GND Pin for the power supply (COM1 GND)

3.5.13 DS-715/PPDS-715-MTCP





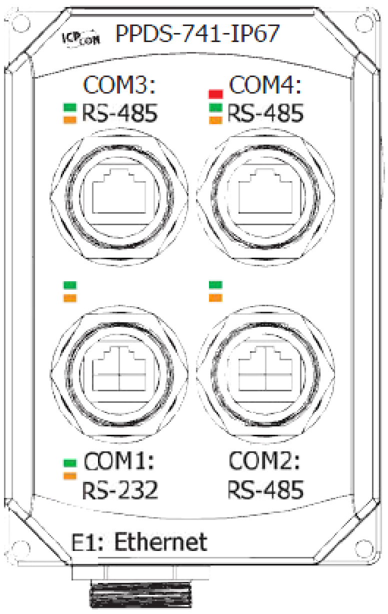
Terminal No.	Pin Assignment
E1	
01	N/A
02	N/A
03	N/A
04	N/A
05	INIT*
06	N/A
07	N/A
08	(R)+Vs
09	(B)GND

COM1 (RS-422/485)
F.G.
Tx+/D+
Tx-/D-
Rx+
Rx-

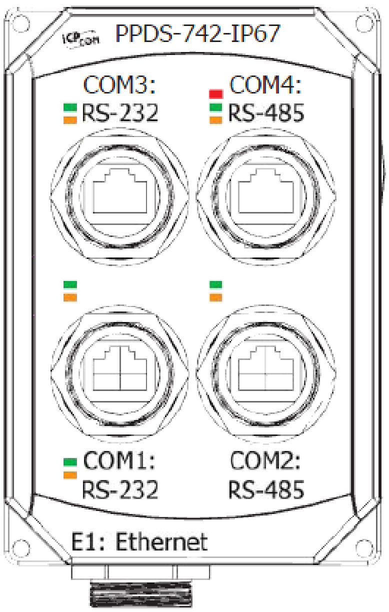
INIT*	Initialization pin (for enabling/disabling AUTOEXEC.BAT)
(R)Vs+	V+ Pin for the power supply (+12 ~ +48 V_{DC} unregulated)
(B)GND	GND Pin for the power supply (COM1 GND)

3.5.14 PPDS-700-IP67

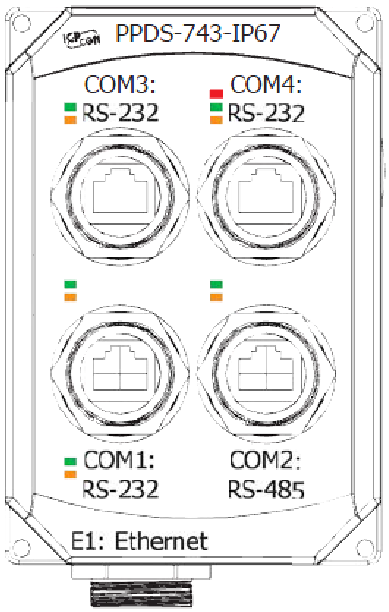
PPDS-741-IP67



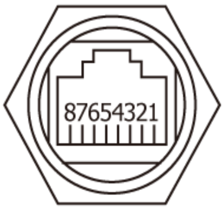
PPDS-742-IP67



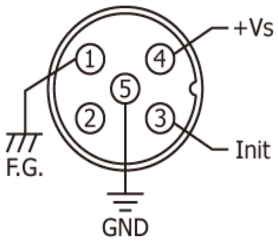
PPDS-743-IP67



COM1 ~ COM4



DC +12 ~ +48 V_{DC}

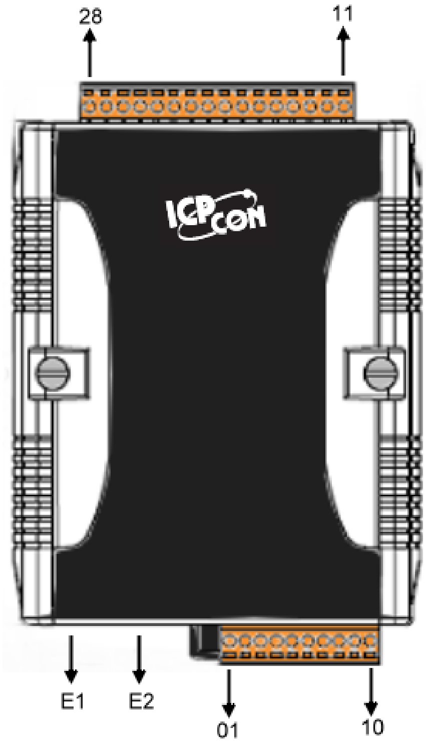


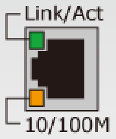
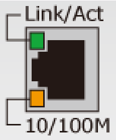
LED Indicators		
System	Red	Sys.
Ethernet	Green	Link/Act (E1)
	Orange	10/100M (E1)
COM1 ~ COM4	Green	RxD
	Orange	TxD

Pin	5-wire RS-232	2-wire RS-485
1	--	--
2	RTS	--
3	GND	GND
4	TxD	--
5	RxD	DATA+
6	--	DATA-
7	CTS	--
8	--	--

Pin	Name
1	F.G.
2	--
3	Init
4	+Vs
5	GND

3.5.15 PDS-5105D-MTCP



Terminal No.		Pin Assignment
E1		 Link/Act 10/100M
E2		 Link/Act 10/100M
COM2	01	D2+
	02	D2-
COM1	03	GND
	04	RxD
	05	TxD
	06	D1+
	07	D1-
	08	PWR
	09	P.GND
	10	F.G.

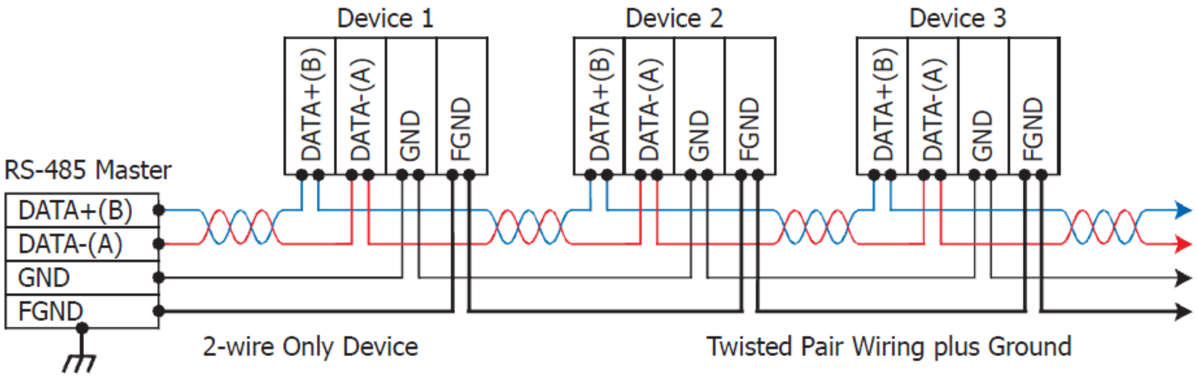
Terminal No.		Pin Assignment
	28	GND
COM10	27	D10-
	26	D10+
COM9	25	D9-
	24	D9+
COM8	23	D8-
	22	D8+
COM7	21	D7-
	20	D7+
COM6	19	D6-
	18	D6+
COM5	17	D5-
	16	D5+
COM4	15	D4-
	14	D4+
COM3	13	D3-
	12	D3+
	11	GND

PWR	V+ pin for the power supply (+12 ~ +48 V_{DC} unregulated)
P.GND	GND pin for the power supply
F.G.	Frame Ground pin

3.6 Wiring Notes

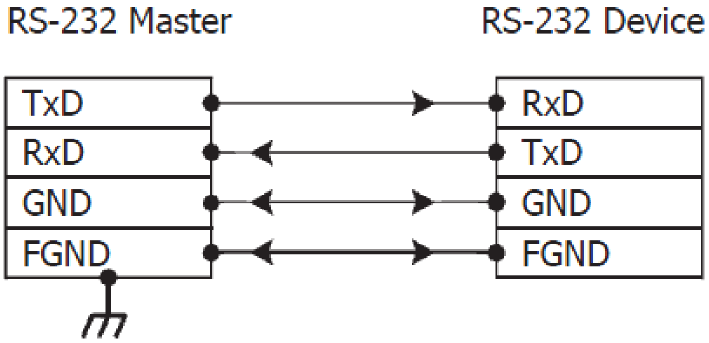
3.6.1 RS-485 Wiring

2-wire RS-485 Wiring

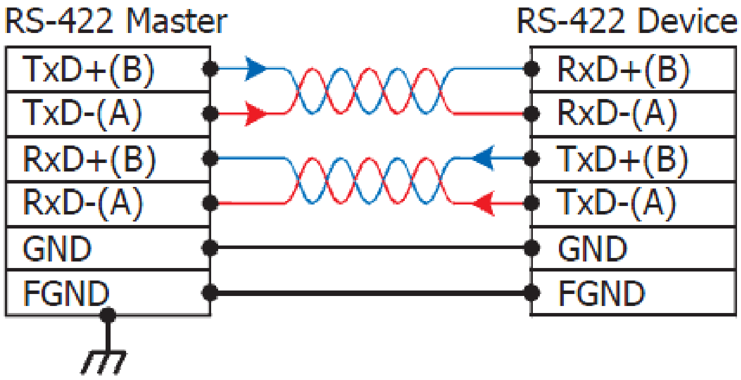


3.6.2 RS-232 Wiring

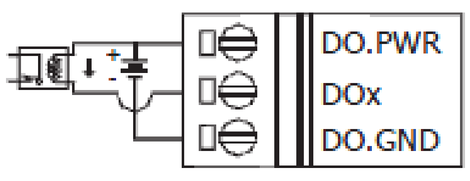
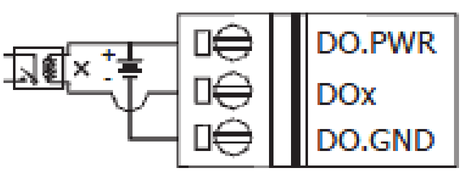
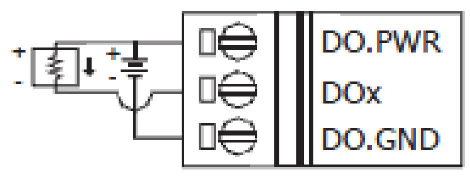
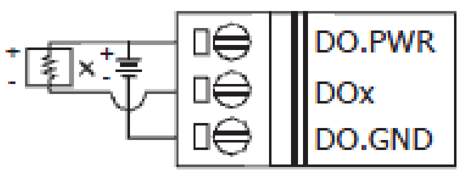
3-wire RS-232 Wiring



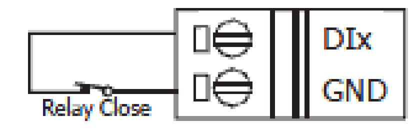
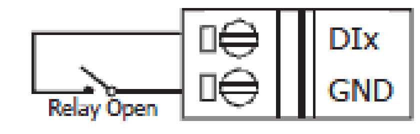
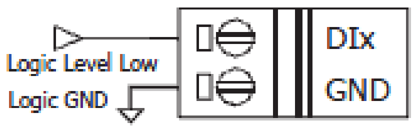
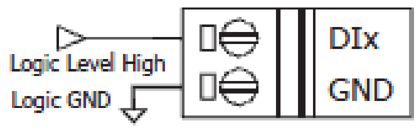
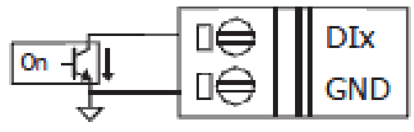
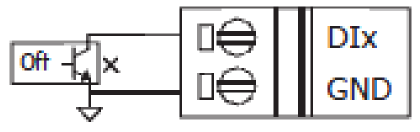
4-wire RS-232 Wiring



3.6.3 Digital Output Wiring

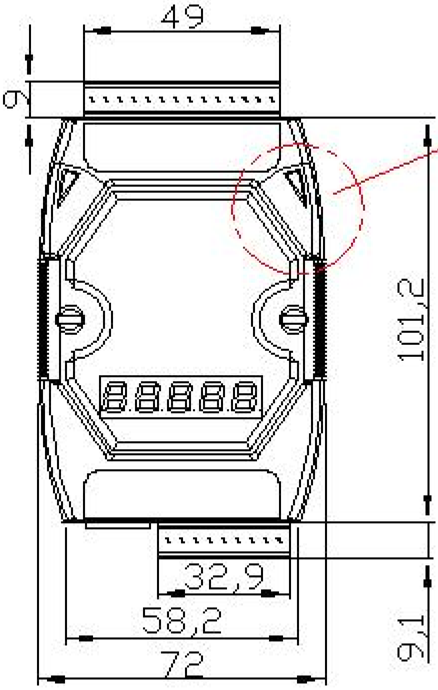
Output Type	DO Command as 1	DO Command as 0
Drive Relay	Relay ON	Relay Off
		
Resistance Load		
		

3.6.4 Digital Input Wiring

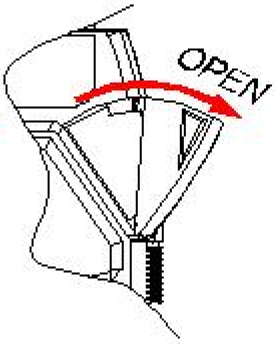
Input Type	DI Value as 0	DI Value as 1
Relay Contact	Relay ON	Relay Off
		
TTL/CMOS Logic	Voltage < 1V	Voltage > 3.5V
		
Open Collector	Open Collector On	Open Collector Off
		

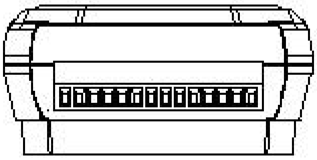
3.7 Dimensions and Mounting

- PDS-700(D)/PPDS-700(D)-MTCP

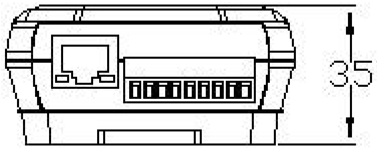


Front View

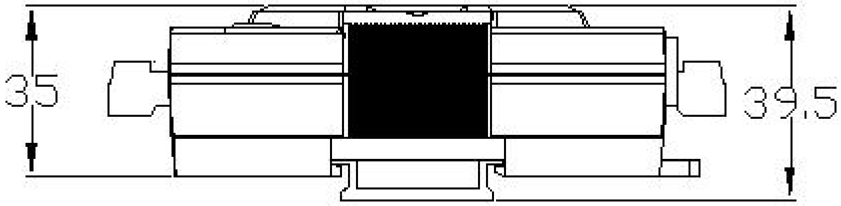




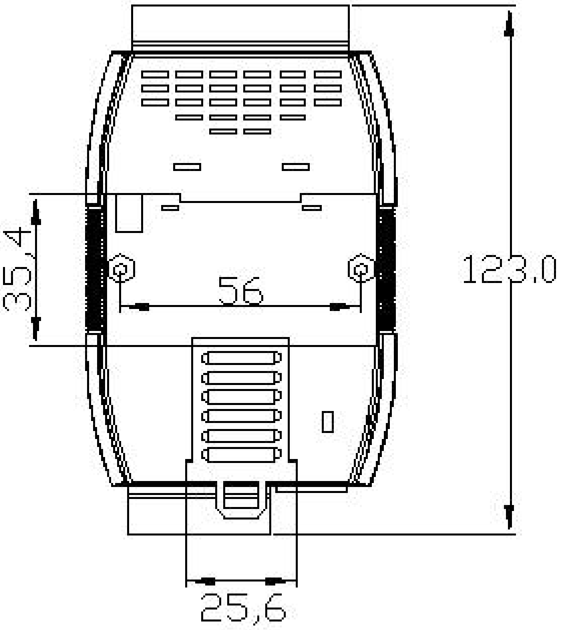
Top View



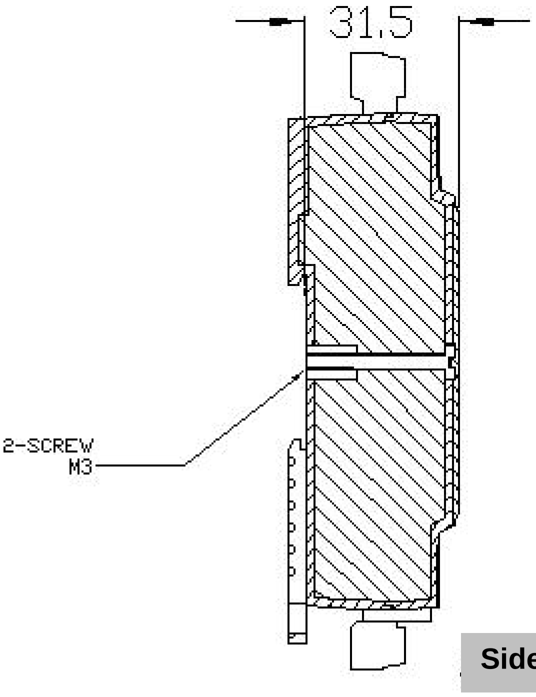
Bottom View



Din-Rail Mounting Bracket

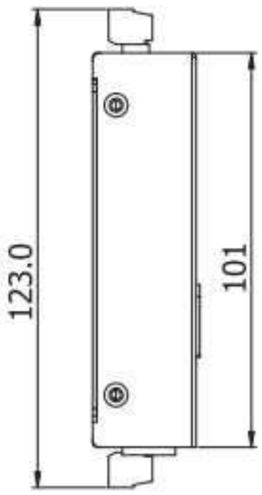


Back View

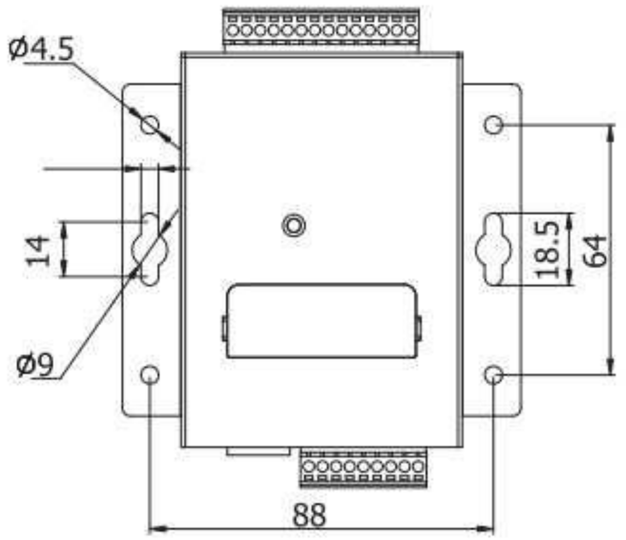


Side View

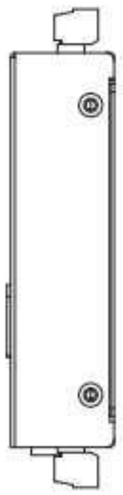
- PDSM-700(D)/PPDSM-700(D)-MTCP



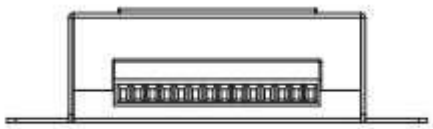
Left Side View



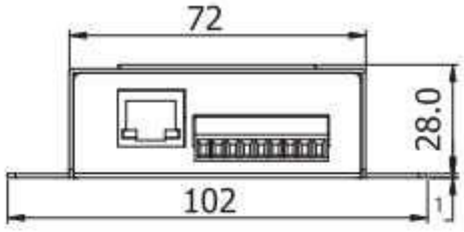
Front View



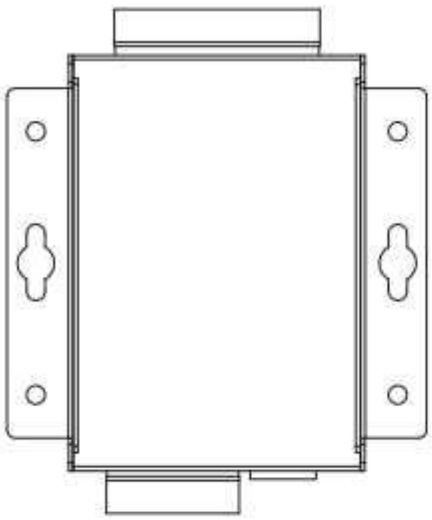
Right Side View



Top View

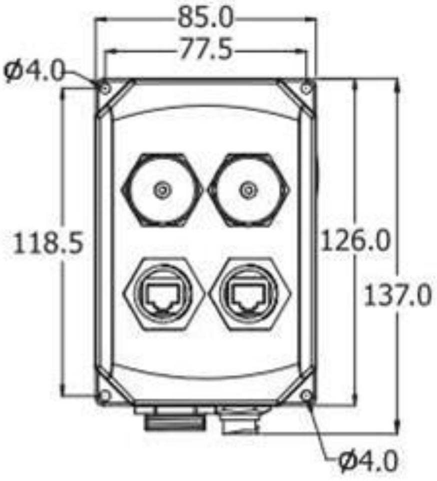


Back View

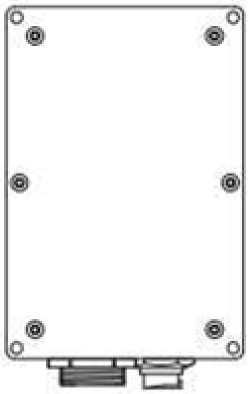


Bottom View

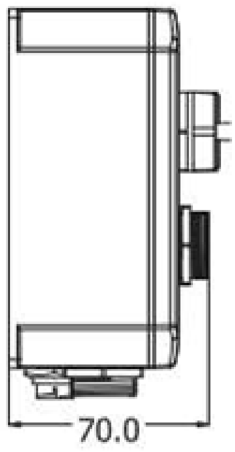
PPDS-700-IP67



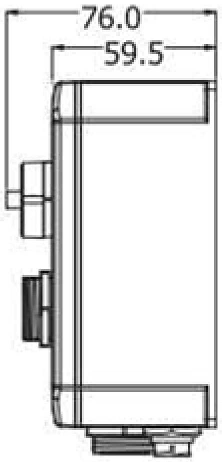
Front View



Back View

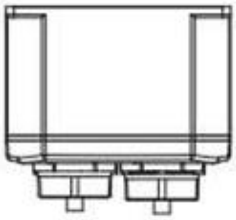


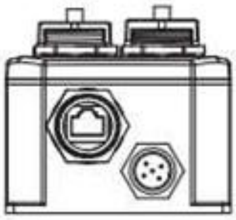
Left Side View



Right Side View

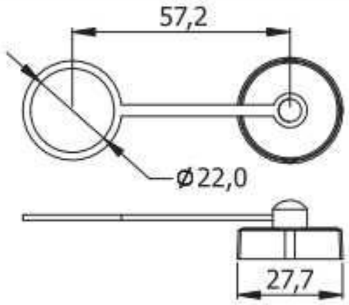
Top View





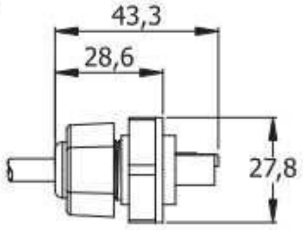
Bottom View

IP67 Ethernet Cap with Tether



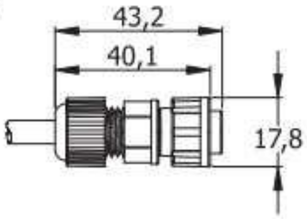
IP67 Ethernet Plug

Cable Dia:
Max. 7.0 mm
Min. 5.5 mm

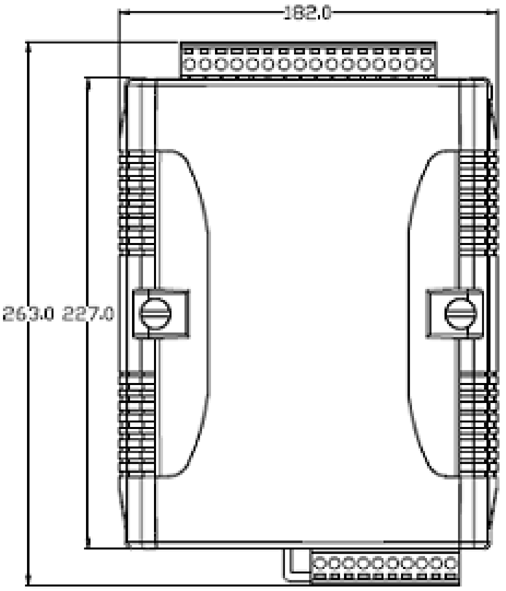


IP67 PWR Plug

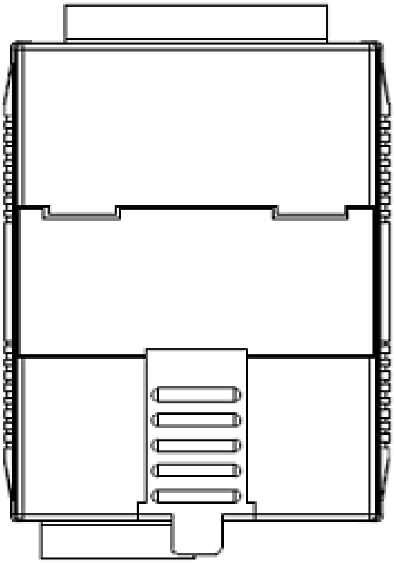
Cable Dia:
Max. 6.5 mm
Min. 5.0 mm



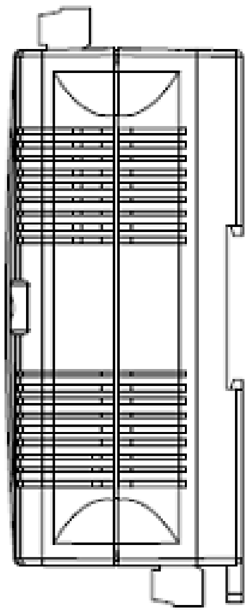
■ PDS-5105D-MTCP



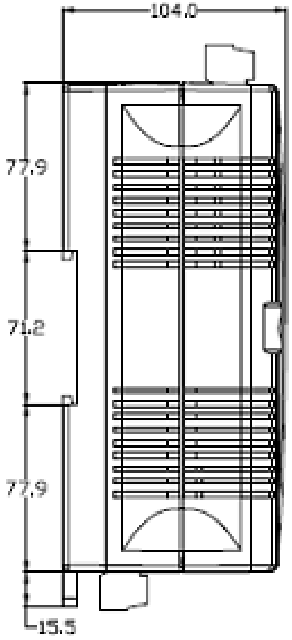
Front View



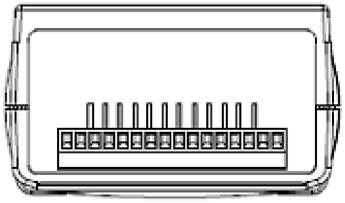
Back View



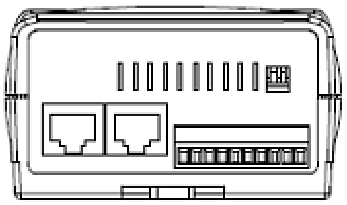
Left Side View



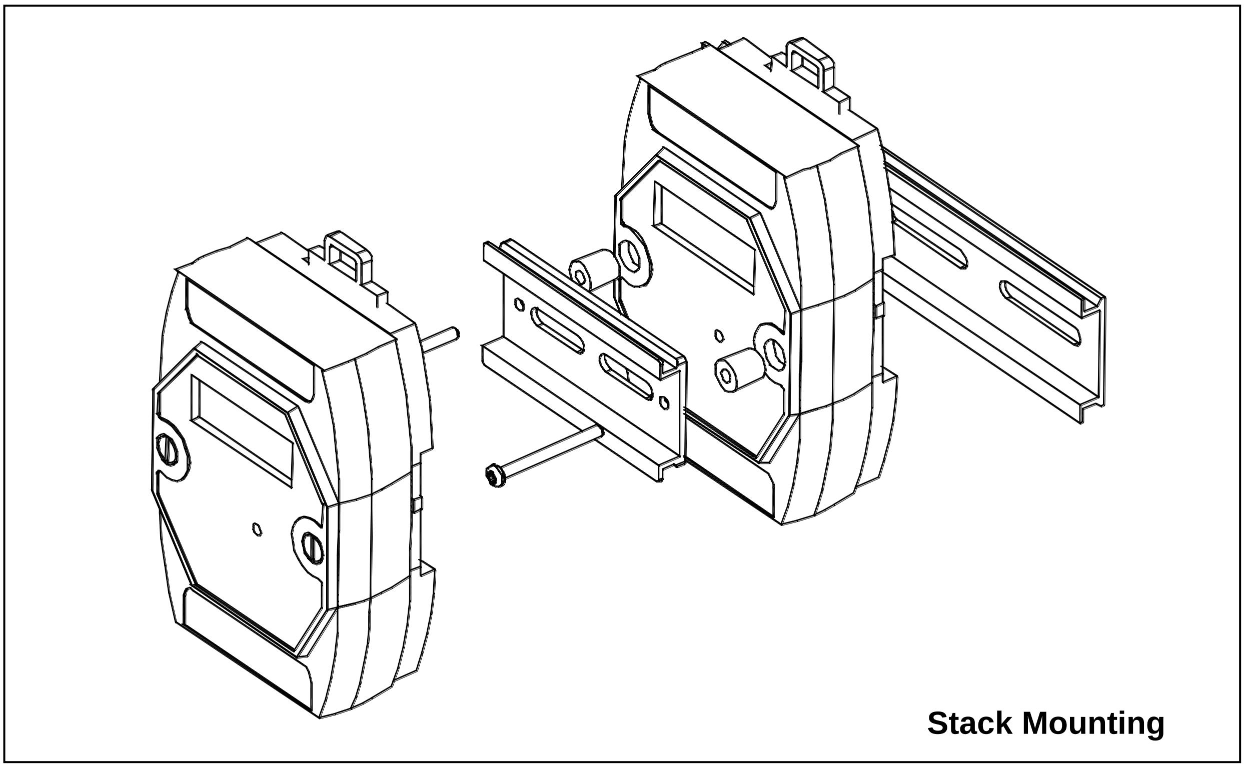
Right Side View



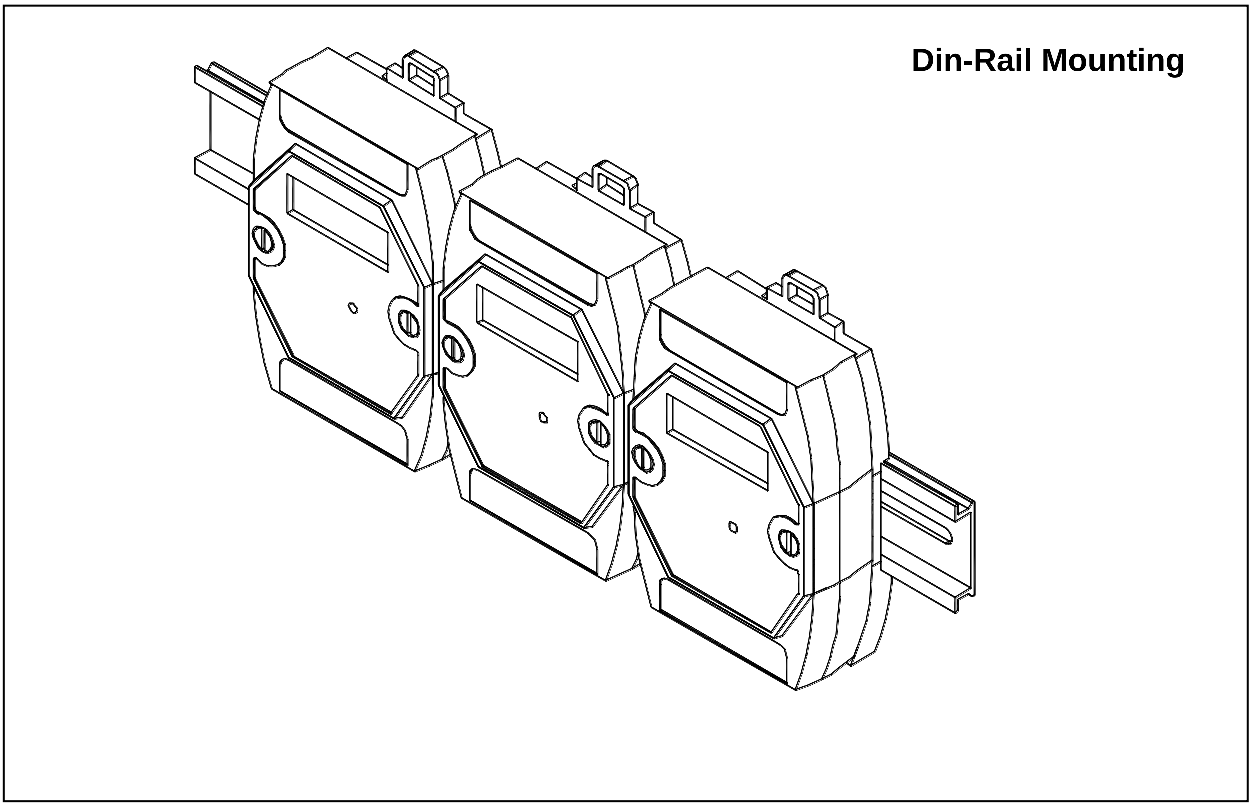
Top View



Bottom View

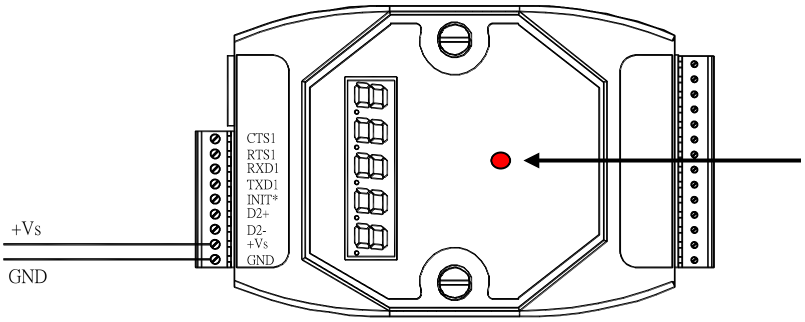


Stack Mounting



Din-Rail Mounting

3.8 PDS Series Diagnostics



Indicator LED

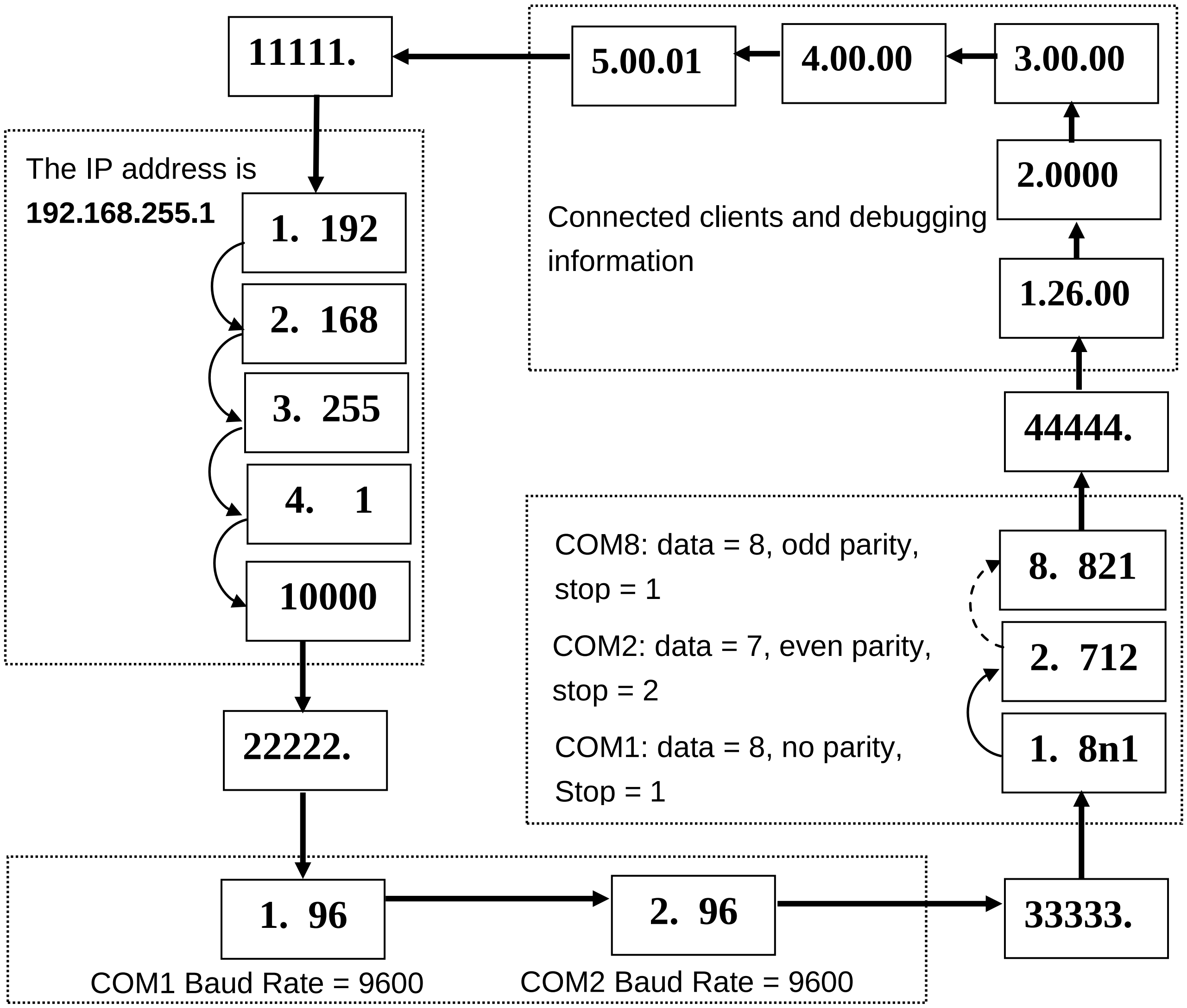
VxComm running: **On/Off**

Xserver running: **On/Off**

Step 1: Apply power (+Vs, GND) to the PDS. The PDS(M)-700 power supply can range from +10 V ~ +30 V. The PPDS(M)-700-MTCP, DS-700, PPDS-700-IP67, PDS-782-25 and PDS-5105D-MTCP power supply can range from +12 V ~ +48 V.

Step 2: Check the 5-digit 7-SEG LED. Data will be shown as follows:

Note: Only D-version modules have a 5-digit 7-SEG LED.



The IP address is **192.168.255.1**

COM1 Baud Rate = 9600

COM2 Baud Rate = 9600

COM8: data = 8, odd parity, stop = 1

COM2: data = 7, even parity, stop = 2

COM1: data = 8, no parity, Stop = 1

Connected clients and debugging information

Information related to the PDS series module can be classified into 4 main areas:

- Group ID 11111: The IP address information for the PDS series
- Group ID 22222: The Baud Rate for all COM Ports
- Group ID 33333: The COM Port configuration
- Group ID 44444: The Connected clients and debugging information for this PDS series module

The format for the PDS IP address information is as follows:

- 5-Digit LED Group ID: 11111
- LED -1: indicator, which can be either 1, 2, 3 or 4
- LED -2~5: IP address
- TCP command port (Default = 10000)
- DHCP Setting: disabled (0)/enabled (1)

The LED will initially display the Group ID, and will then display the IP address as illustrated in the previous diagram. If the IP address is changed, the value displayed will change immediately. The default shipping IP is 192.168.255.1 and the display sequence is shown in the previous diagram.

The format for the COM Port Baud Rate information is follows:

- 5-Digit LED Group ID : 22222
- LED-1: COM Port number
- LED-2~5: The Baud Rate determined as (Baud Rate/100)

LED-1 displays the COM Port number, with LED-2~5 showing the Baud Rate for that COM Port. The Baud Rate = (value shown by LED-2~5) * 100. Therefore, a COM Port value displayed as 1.96 means that the Baud Rate of COM1 = 9600 bps; a value displayed as 2.1152 means that the Baud Rate of COM2 = 115200 bps. The Baud Rates for all PDS COM Ports will be shown in sequence.

The format for the COM Port configuration information is as follows:

- 5-digit LED Group ID: 33333
- LED-1: COM Port number
- LED-3: Data Bit: 5 , 6 , 7 or 8
- LED-4: Parity Bit, n = no parity, E = even parity, O = odd parity, M = mark parity, S = space parity
- LED-5: Stop Bit: 1 or 2

The format for the connected clients and debugging information is as follows:

- 5-digit LED Group ID: 44444
- LED-1 will display 1, 2, 3, 4, 5 and the module name in sequence.
- When LED-1 is 1, LED-2/3 indicates the number of available free sockets (default is 26 for PDS), and LED-4/5 shows the number of sockets being used by clients (default is 0), e.g. 12600
- When LED-1 is 2, LED-2~5 indicates how many times the PDS-700 has been reset, e.g. 20002 (The PDS has been reset 2 times)
- When LED-1 is 3, the display indicates how many Ethernet packets are currently being received by the PDS.
- When LED-1 is 4, the display indicates that the status of the internal Flag used to allow Ethernet packets to be sent is 0 or 1.
- When LED-1 is 5, the display indicates the number of times the Ethernet chip has been reset.
- Module Name: dS.7xx

When the PDS is first powered-up or if it has just been reset, the reset state = 1. If any client connects to the PDS, the reset state will be changed to 0. In addition, the number of free sockets will be decreased and the number of used sockets will be increased. If the number of free sockets is reduced to 0, then no additional clients will be able to link to the PDS. The default number of free-sockets for the PDS is 26. Therefore, the server (VxComm firmware or Xserver firmware) allows 26 connections to be linked to a single PDS. Each client program occupies at least 2 connections for a single serial port, one connection for data and another for commands.

If the 5-digit LEDs do not display the above detail, take the following steps:

- Power off the module
- Connect the INIT* pin to the Vs+ pin
- Power on the module and double check the configuration

Step 3: The red LED on the PDS is used to indicate the following:

- 🔴 → ON
- ⚫ → OFF
- Unit → 0.5 second

OS	→	🔴	⚫	🔴	⚫	🔴	⚫	🔴	⚫	🔴	⚫
M0	→	🔴	⚫	⚫	⚫	⚫	⚫	⚫	⚫	⚫	⚫
M1	→	🔴	⚫	🔴	⚫	⚫	⚫	⚫	⚫	⚫	⚫
M2	→	🔴	⚫	🔴	⚫	🔴	⚫	⚫	⚫	⚫	⚫

The PDS contains either Xserver or VxComm as default when shipped that is in the OS mode.

If the LED is always ON, take the following steps:

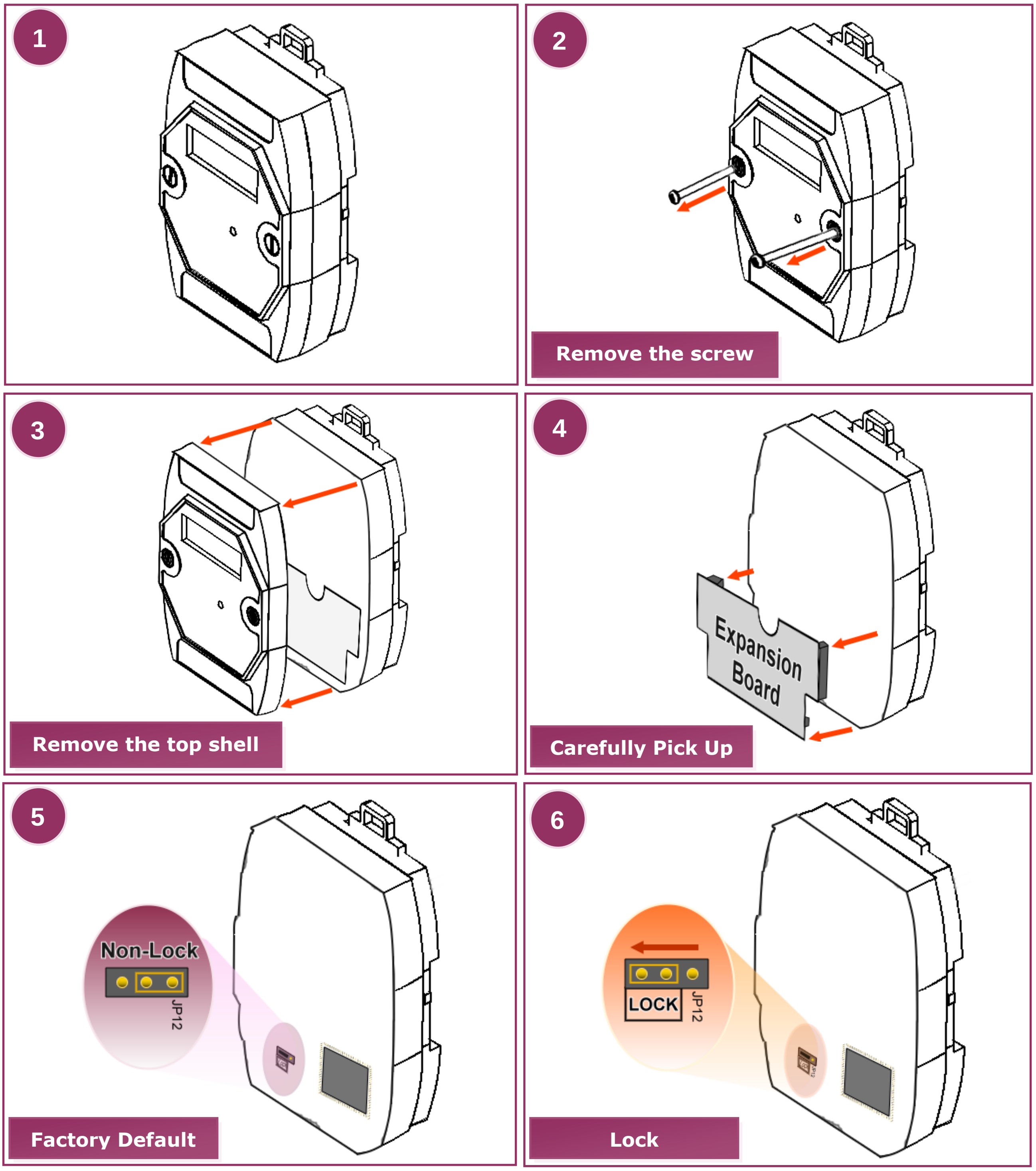
- Power off the module
- Connect the INIT* pin to the Vs+ pin
- Power on the module and double check the configuration

Step 4: Power off the module.

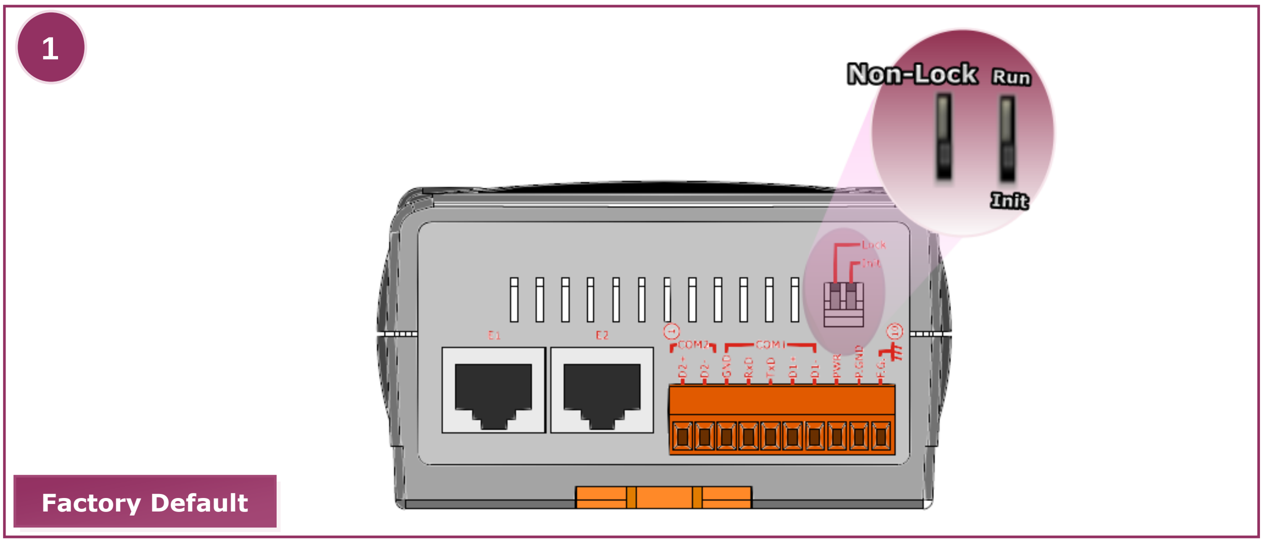
3.9 Flash Protection Function

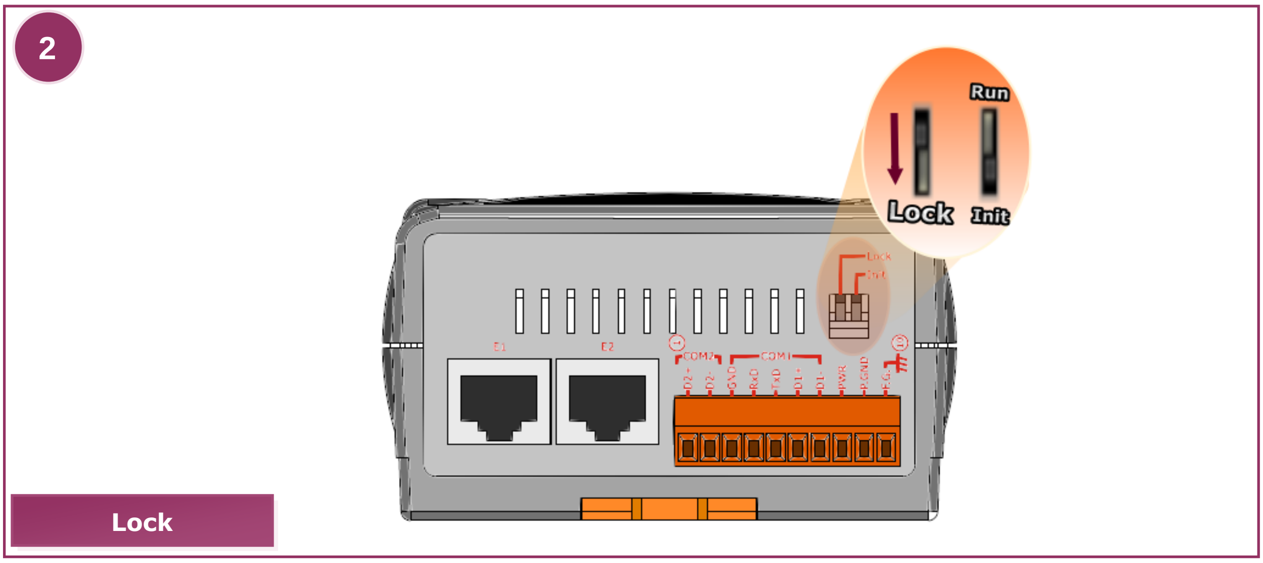
In factory default settings, the “flash protection” function is disabled. User can modify the program by yourself or update the firmware. But if need to avoid writing or erasing any file in the flash, user have to enable the “flash protection” function in the PDS modules. Please switch the position of the jumper to “LOCK” and then the PDS modules can restrict any write to flash memory. For more detailed steps as follows:

- **PDS-700 series module:**



PDS-5105D-MTCP:



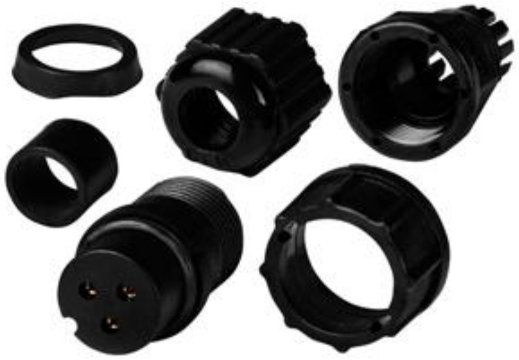


3.10 Waterproofing Attachment Installation for PPDS-700-IP67 Series

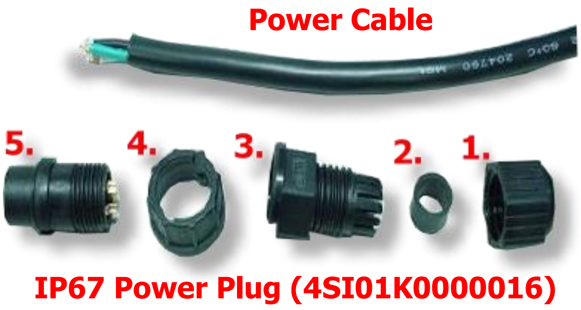
3.10.1 Power Plug Installation

Installation Steps:

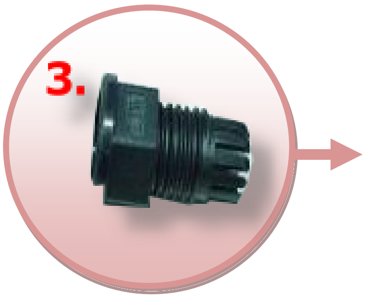
IP67 Power Plug (4SI01K0000016)



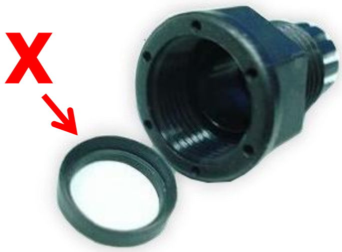
Step 1: Prepare a Power Cable



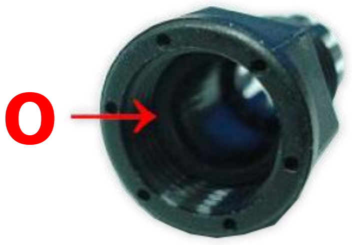
Step 2: Check the item 3 of IP67 power plug



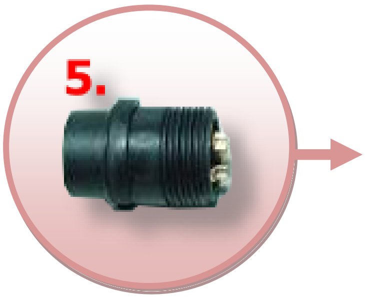
Wrong



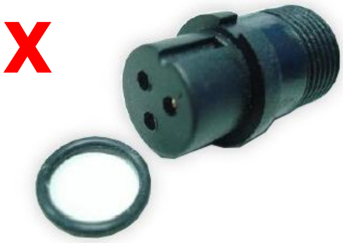
Correct



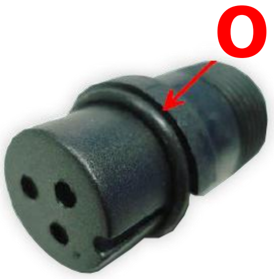
Step 3: Check the item 5 of IP67 power plug



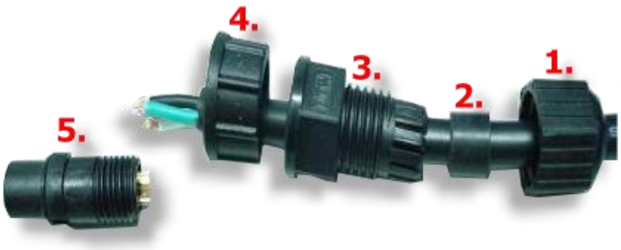
Wrong



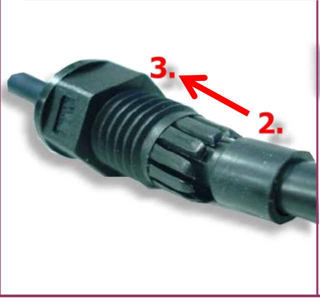
Correct

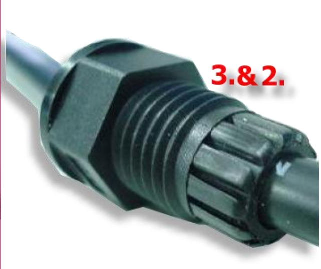


Step 4

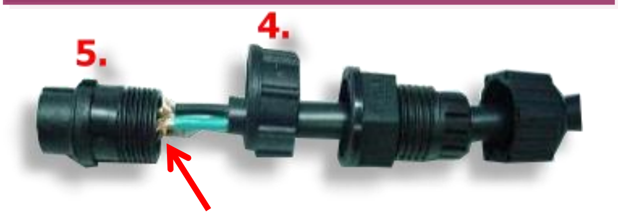


Step 5: Assemble item 2 and 3



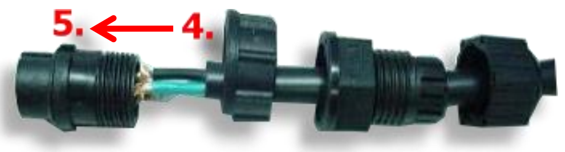


Step 6: Weld power plug and wires

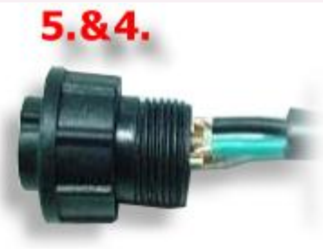


Welding (see page 45 for pin-assignment)

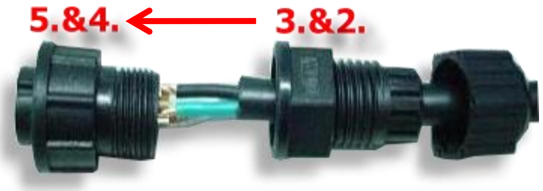
Step 7: Assemble item 4 and 5



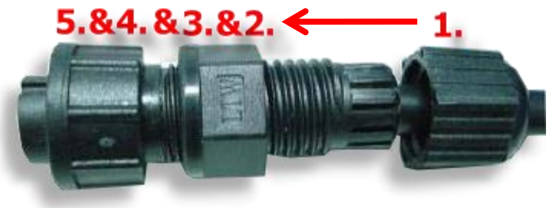
Step 8: Assemble Power Plug



Step 9: Assemble item 3&2 and 5&4

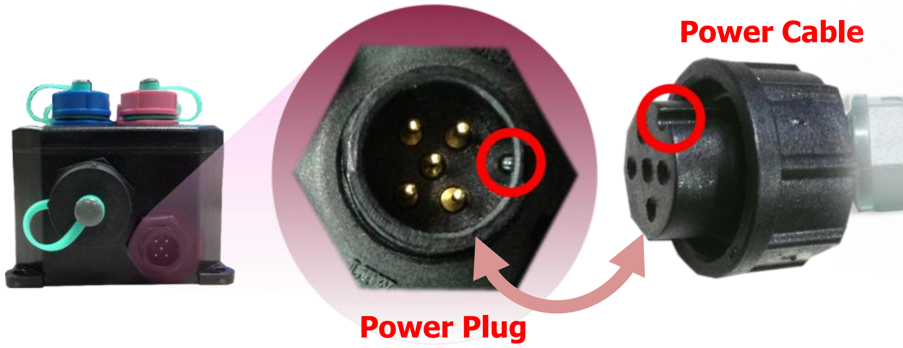


Step 10: Assemble item 1 and 5&4&3&2



Step 11: Make sure to tighten firmly

Step 12: Connect the Power cable to PPDS-700-IP67 on Power plug



The fool-proofing groove (as red circle) is useful for easy connection of power cable and power plug.

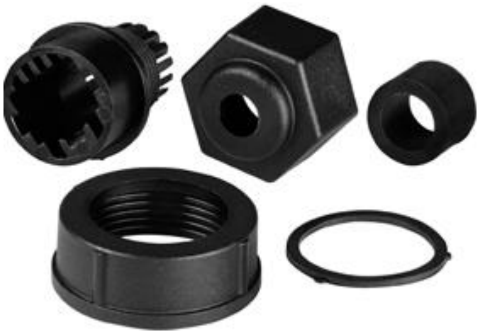
Please make sure they are located in the same direction when connecting these two items.

-Complete-

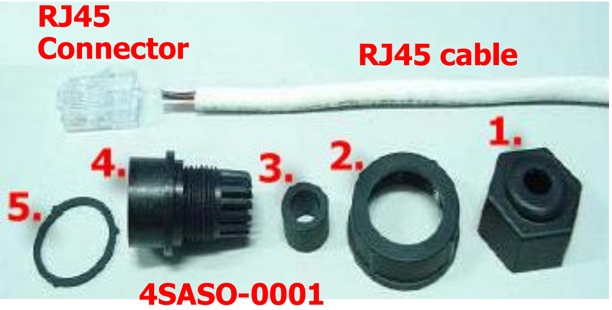
3.10.1 RJ45 Plug Installation

Installation Steps:

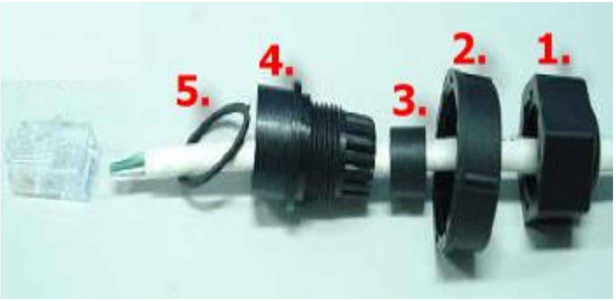
IP67 RJ45 Plug (4SASO-0001)



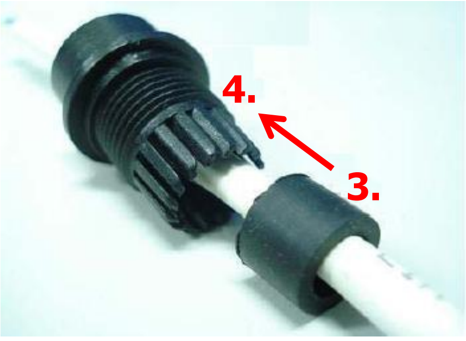
Step 1: Prepare a RJ45 Cable



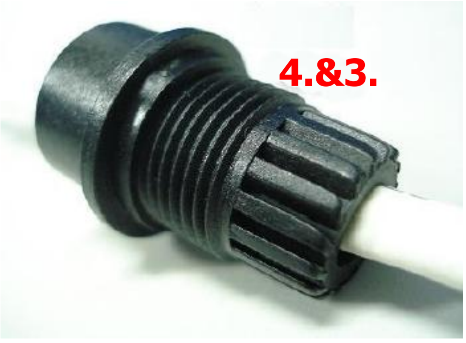
Step 2



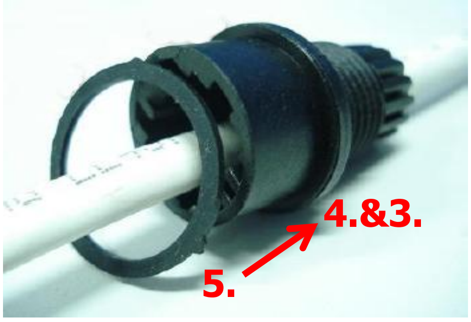
Step 3: Assemble item 3 and 4



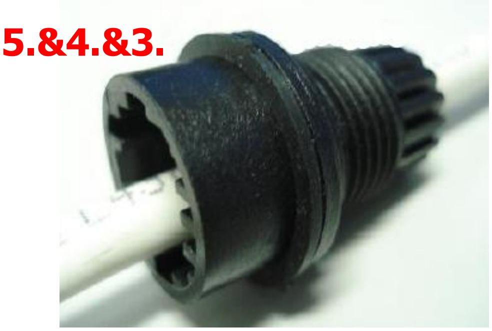
Step 4



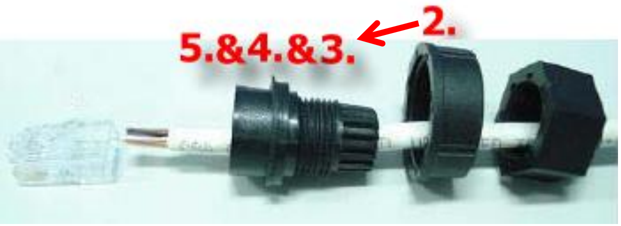
Step 5: Assemble item 3&4 and 5



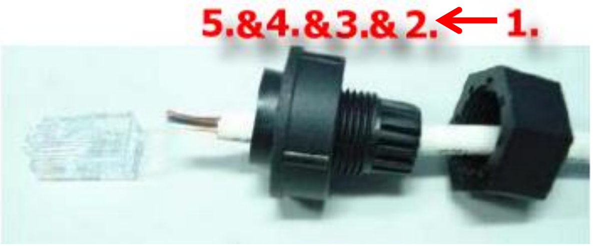
Step 6



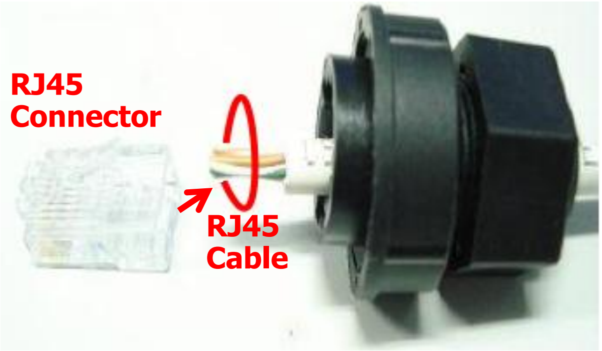
Step 7: Assemble item 2 and 3&4&5



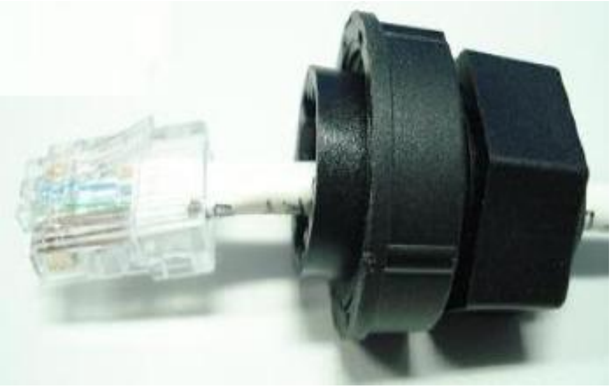
Step 8: Assemble item 1 and 2&3&4&5



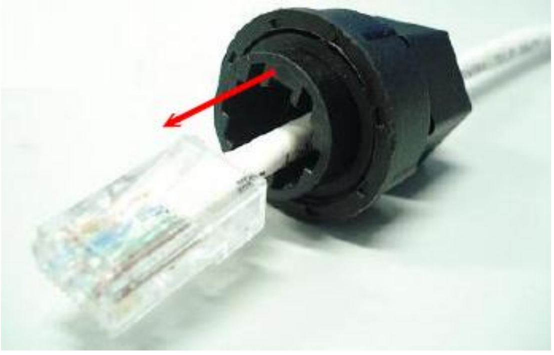
Step 9: Assemble item RJ45 connector and RJ45 cable (See page 45 for pin-assignment)



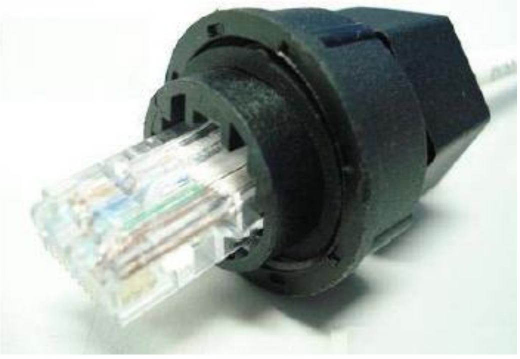
Step 10



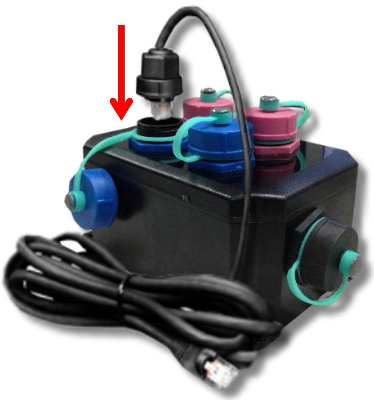
Step 11: Assemble RJ45 Plug



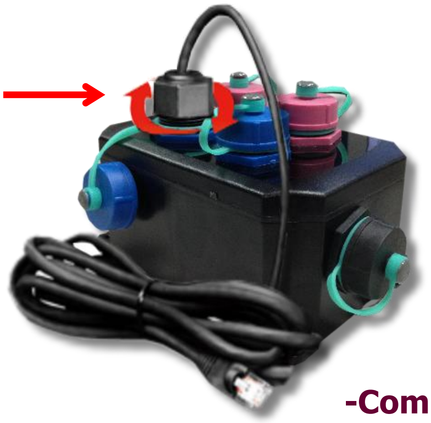
Step 12: Make sure to tighten firmly



Step 13: Connect the RJ45 Cable to PPDS-700-IP67 on COM Port



Step 14: Make sure to tighten firmly



-Complete-

4. Setting up the PDS module

Step 1: Connect the PDS module to the Ethernet Network

Before connecting the PDS module to an Ethernet network, the following items are needed:

1. Power Supply: **+10 ~ +30 V_{DC}** for PDS(M)-700
+12 ~ +48 V_{DC} for DS-700, PPDS(M)-700-MTCP, PPDS-700-IP67, PDS-782-25 and PDS-5105D-MTCP
(eg: DP-665: http://www.icpdas.com/products/Accessories/power_supply/power_list.htm)

2. Hub (eg: NS-205 http://www.icpdas.com/products/Switch/industrial/industrial_list.htm)

3. The network settings in the PC are correctly configured and the Ethernet connection is functioning normally.

4. Disable or correctly configure the Windows firewall and any Anti-Virus software firewall first or else the "**Search Servers**" function in the VxComm Utility may not work. (Contact your System Administrator for more details of how to do this.)

5. Connect the PDS series module to the Ethernet as shown on the following page and switch on the power.

6. Make sure the indicator LED is flashing.

If your PDS series module is a **D**-version module, the 5-digit 7-SEG LED will be used to indicate the system information described in Sec. 3.8

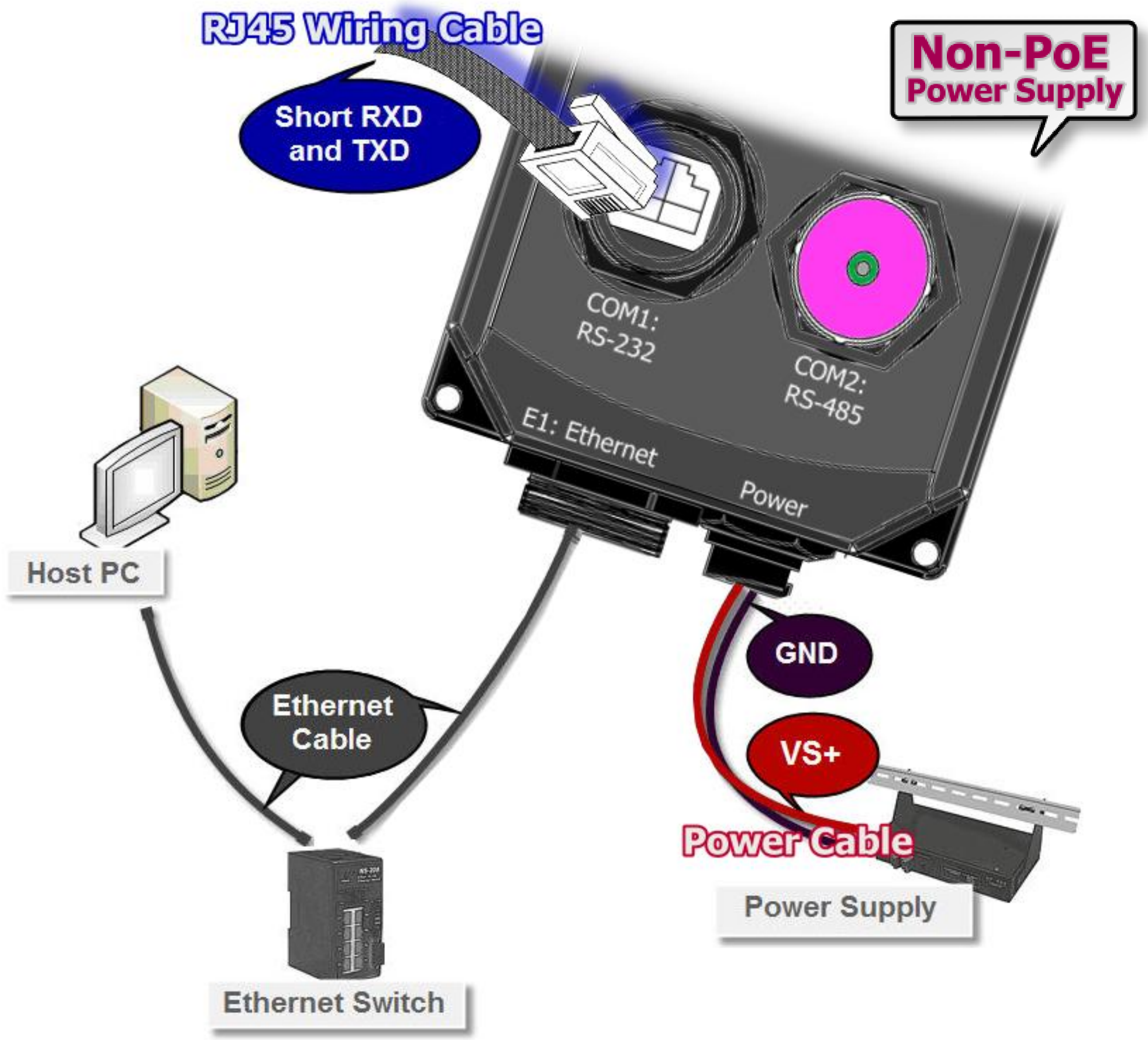
7. Install VxComm Utility on your PC.
The software is located at:
CD: \Napdos\Driver\VxComm_Driver\
http://ftp.icpdas.com/pub/cd/8000cd/napdos/driver/vxcomm_driver/

For wiring diagram as follows:

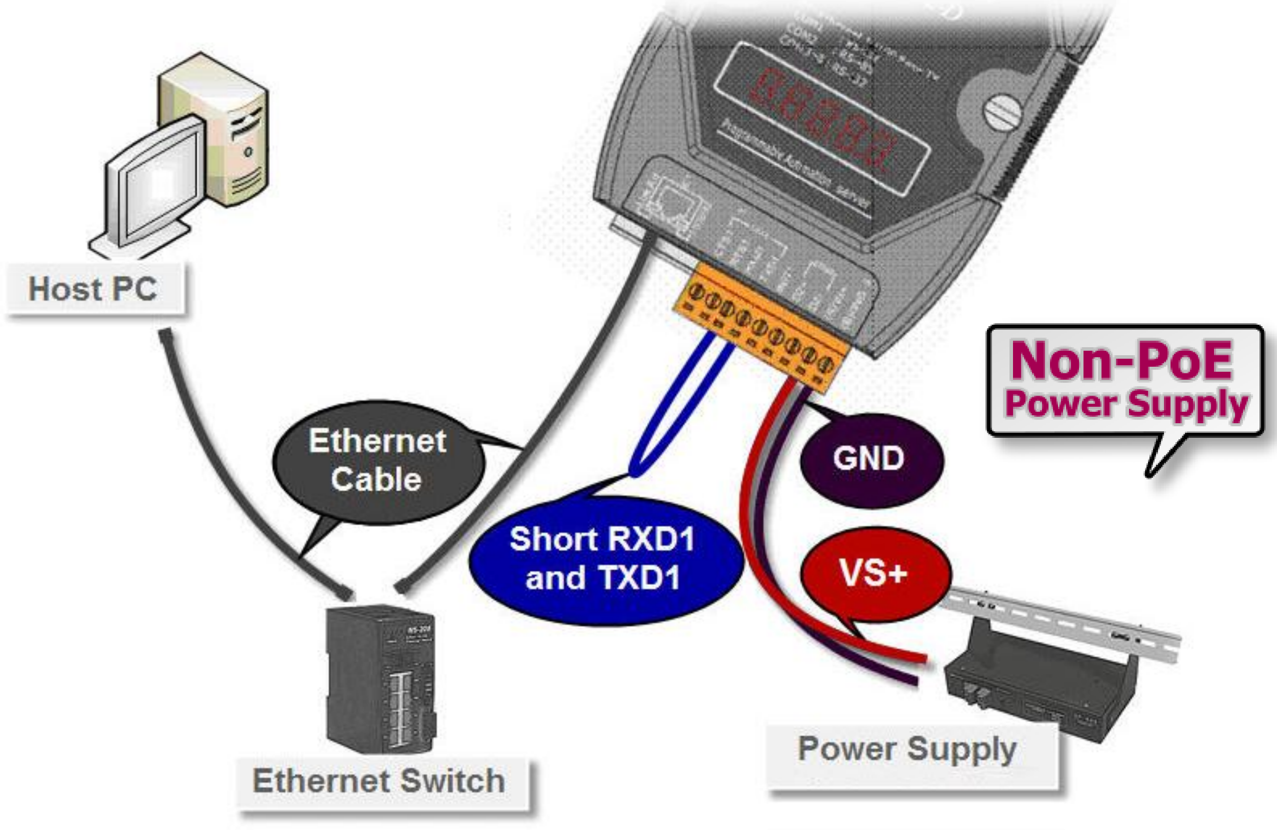
1. Connect both the PDS series module and your computer to the same sub network or the same Ethernet Switch.
2. Short the **RxD and TxD pins** of the PDS series module for execute a self-test.
3. Supply **24 V$_{DC}$ (10 ~ 30 V$_{DC}$) power** to the PDS(M)-700 module.
   Supply **24 V$_{DC}$ (12 ~ 48 V$_{DC}$) power** to the PPDS(M)-700-MTCP, DS-700, PPDS-700-IP67, PDS-782-25 and PDS-5105D-MTCP module.

■ **PPDS-700-IP67 series:**

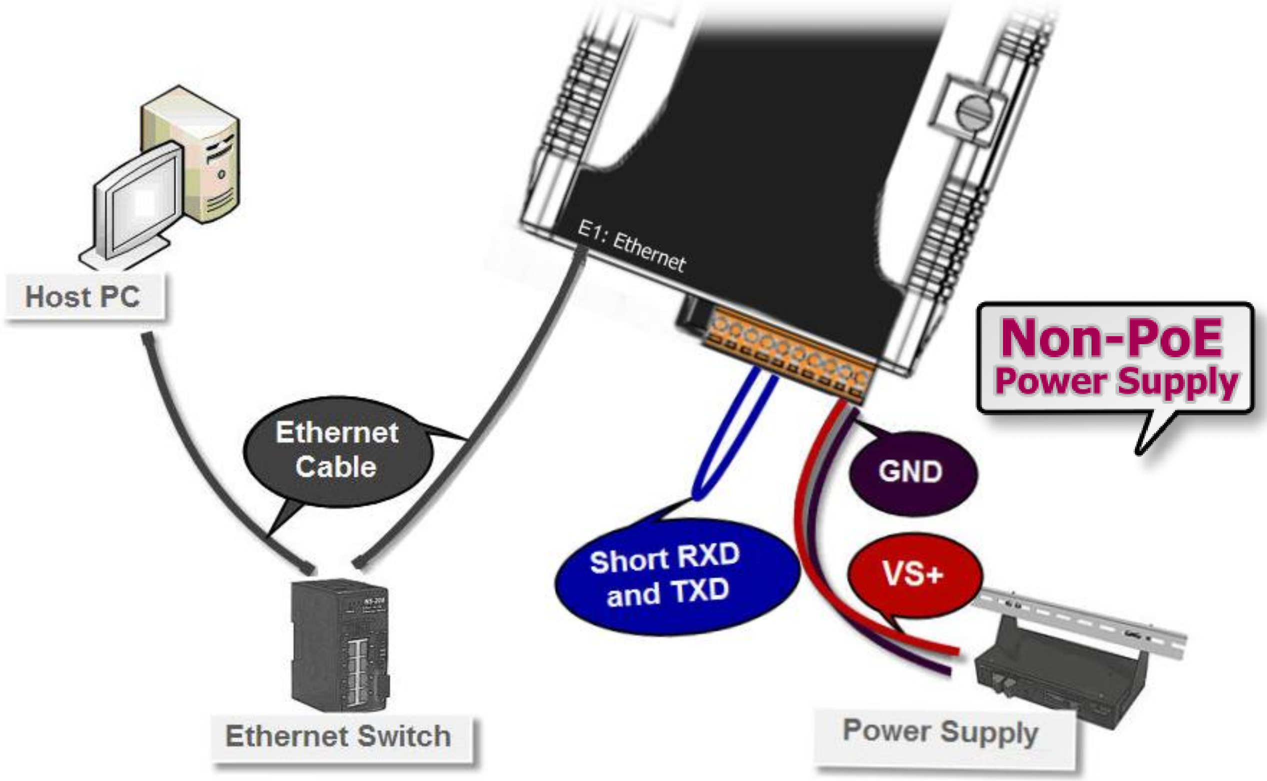
Note: Please refer to the pin definitions of PPDS-700-IP67 (Sec. 3.5.14) to make RJ45 wiring cable and power cable.



- **DS-700/PDS(M)-700(D)/PPDS(M)-700(D)-MTCP series:**



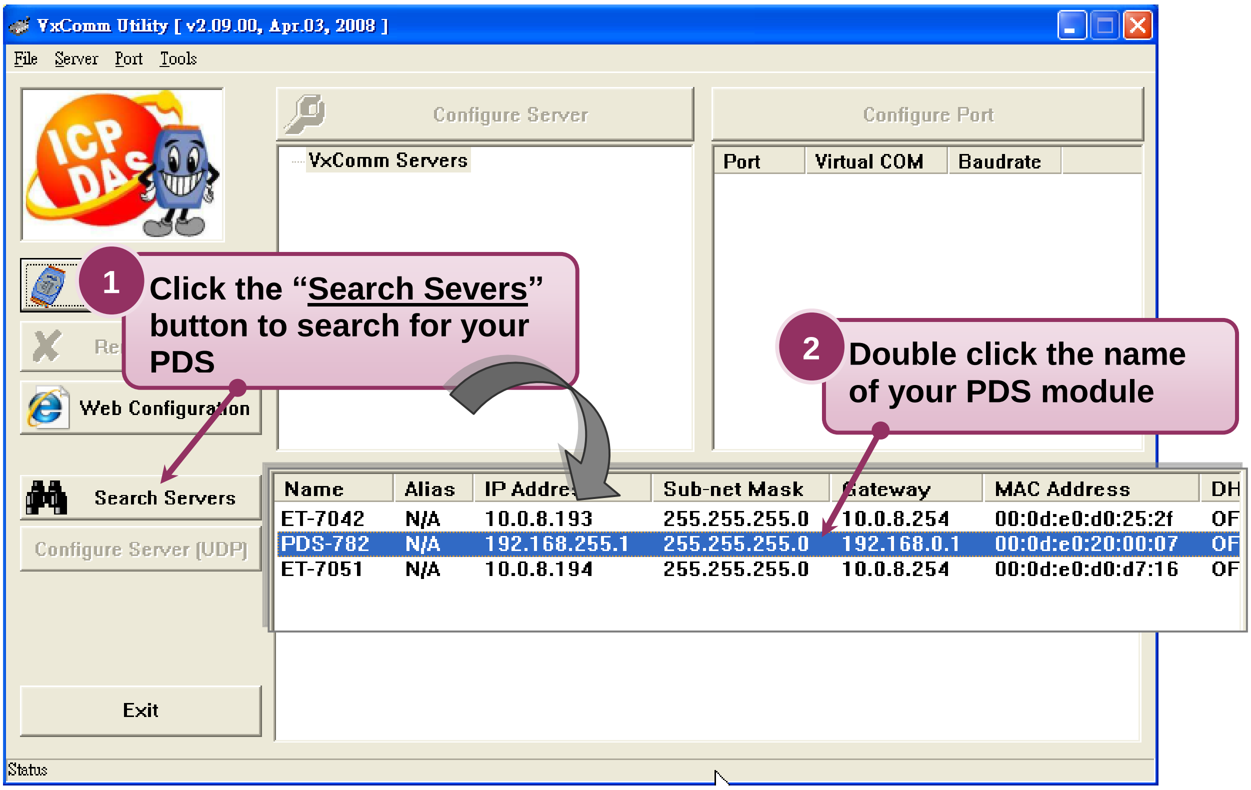
- **PDS-5105D-MTCP:**



Step 2: Search for the PDS module on the Ethernet network

1. Execute the VxComm Utility and then search for your PDS series module.

2. Double click the name of the PDS to open the configuration settings dialog box.

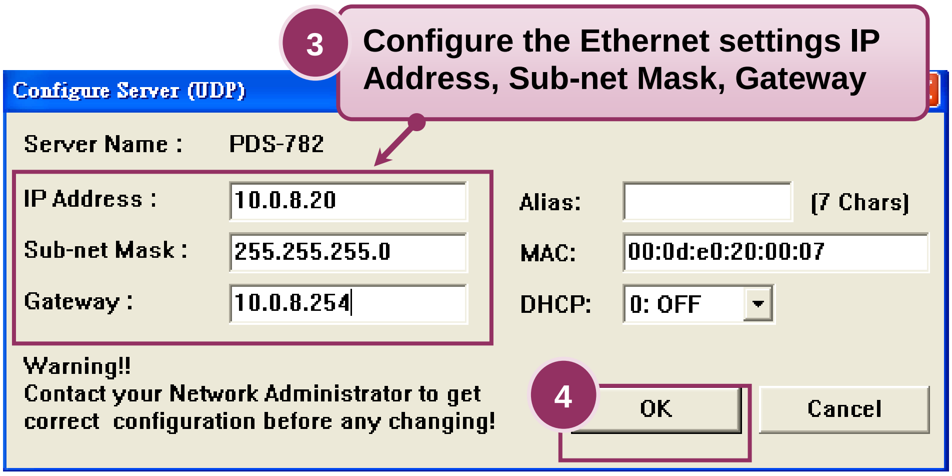


1 Click the "Search Severs" button to search for your PDS

2 Double click the name of your PDS module

Name	Alias	IP Addre	Sub-net Mask	Gateway	MAC Address	DH
ET-7042	N/A	10.0.8.193	255.255.255.0	10.0.8.254	00:0d:e0:d0:25:2f	OF
PDS-782	N/A	192.168.255.1	255.255.255.0	192.168.0.1	00:0d:e0:20:00:07	OF
ET-7051	N/A	10.0.8.194	255.255.255.0	10.0.8.254	00:0d:e0:d0:d7:16	OF

3. Contact your Network Administrator to obtain the correct network configuration details (such as IP/Mask/Gateway). Enter the network settings and then click "**OK**". The PDS series module will be restarted itself immediately.



3 Configure the Ethernet settings IP Address, Sub-net Mask, Gateway

Server Name : PDS-782

IP Address : 10.0.8.20

Sub-net Mask : 255.255.255.0

Gateway : 10.0.8.254

Alias: (7 Chars)

MAC: 00:0d:e0:20:00:07

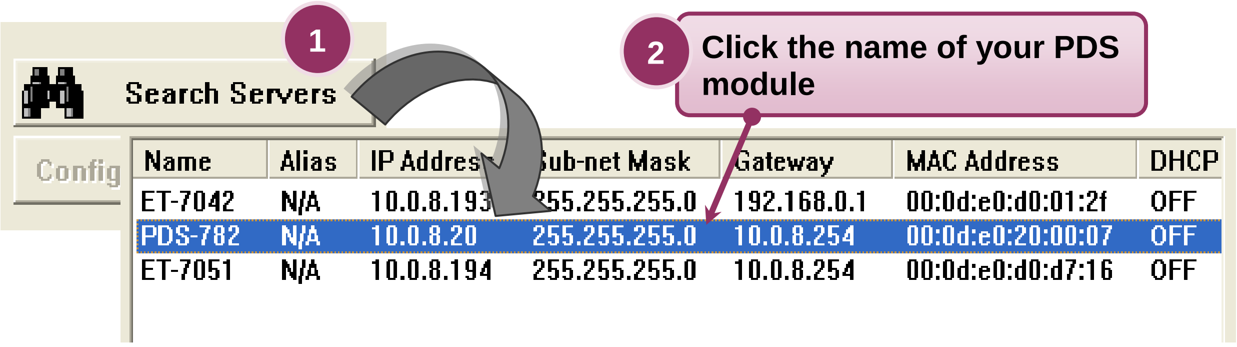
DHCP: 0: OFF

Warning!!
Contact your Network Administrator to get correct configuration before any changing!

4

Step 3: Configuring Virtual COM Ports

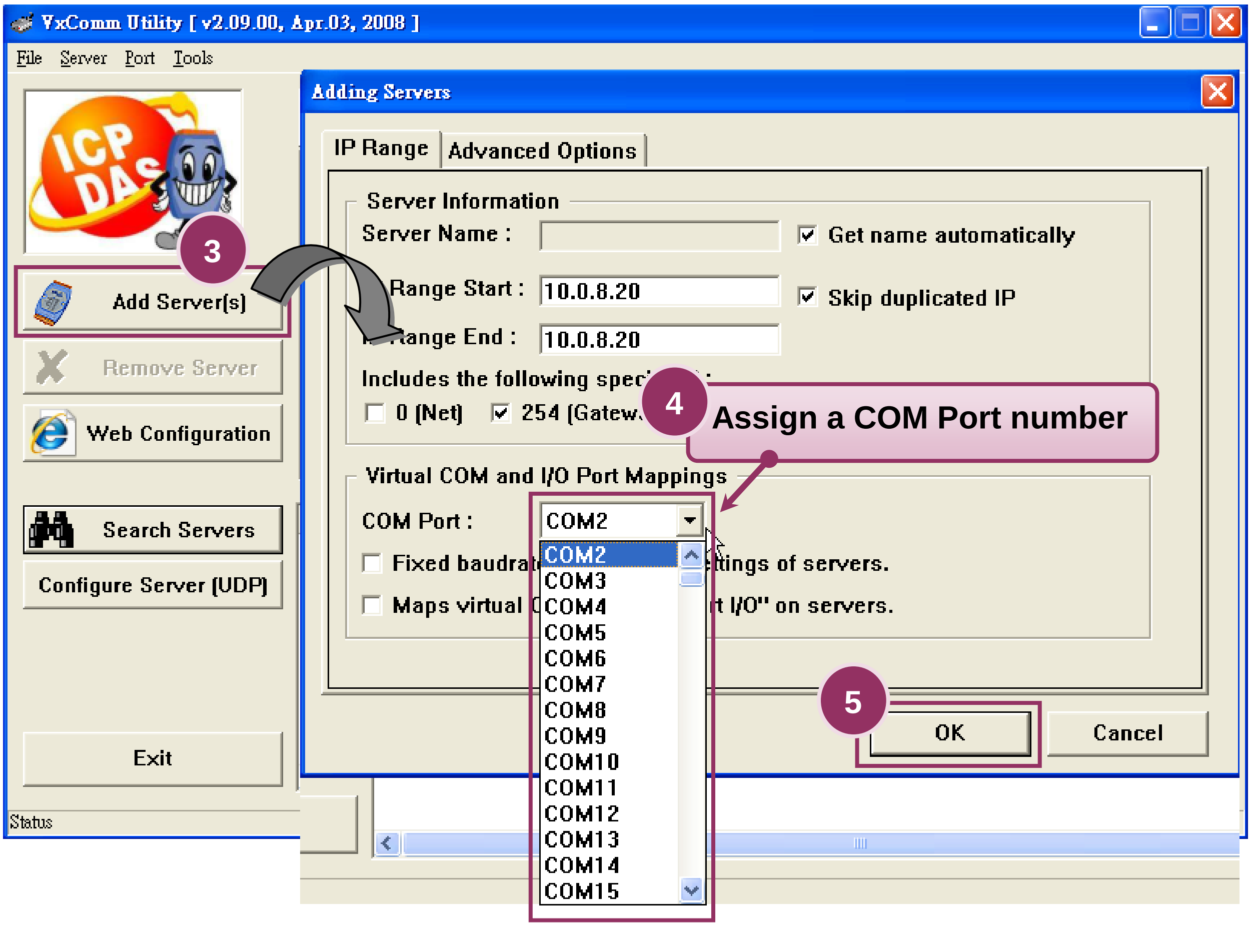
1. Click the "**Search Servers**" button again to search for your PDS series module to make sure that the new IP/Mask/Gateway settings have been saved, then click the name of your PDS series module once to select it.



Click the name of your PDS module

Name	Alias	IP Addres	ub-net Mask	Gateway	MAC Address	DHCP
ET-7042	N/A	10.0.8.193	255.255.255.0	192.168.0.1	00:0d:e0:d0:01:2f	OFF
PDS-782	N/A	10.0.8.20	255.255.255.0	10.0.8.254	00:0d:e0:20:00:07	OFF
ET-7051	N/A	10.0.8.194	255.255.255.0	10.0.8.254	00:0d:e0:d0:d7:16	OFF

2. Click the "**Add Server[S]**" button, then assign a COM Port number and click "**OK**" to save your settings.



Server Information

Server Name : ☑ Get name automatically

Range Start : 10.0.8.20 ☑ Skip duplicated IP

Range End : 10.0.8.20

Includes the following spec

☐ 0 (Net) ☑ 254 (Gatew

Virtual COM and I/O Port Mappings

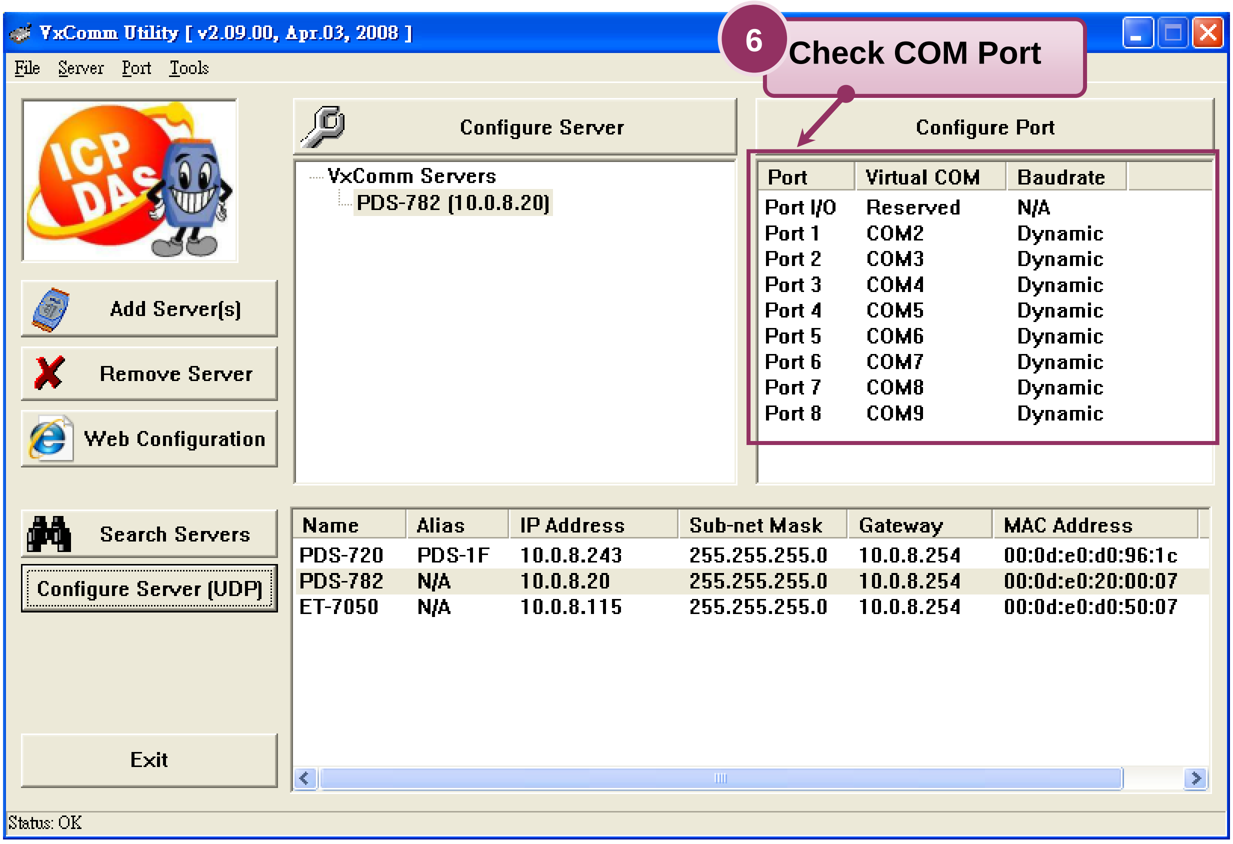
COM Port : COM2

☐ Fixed baudrat ttings of servers.

☐ Maps virtual t I/O" on servers.

Assign a COM Port number

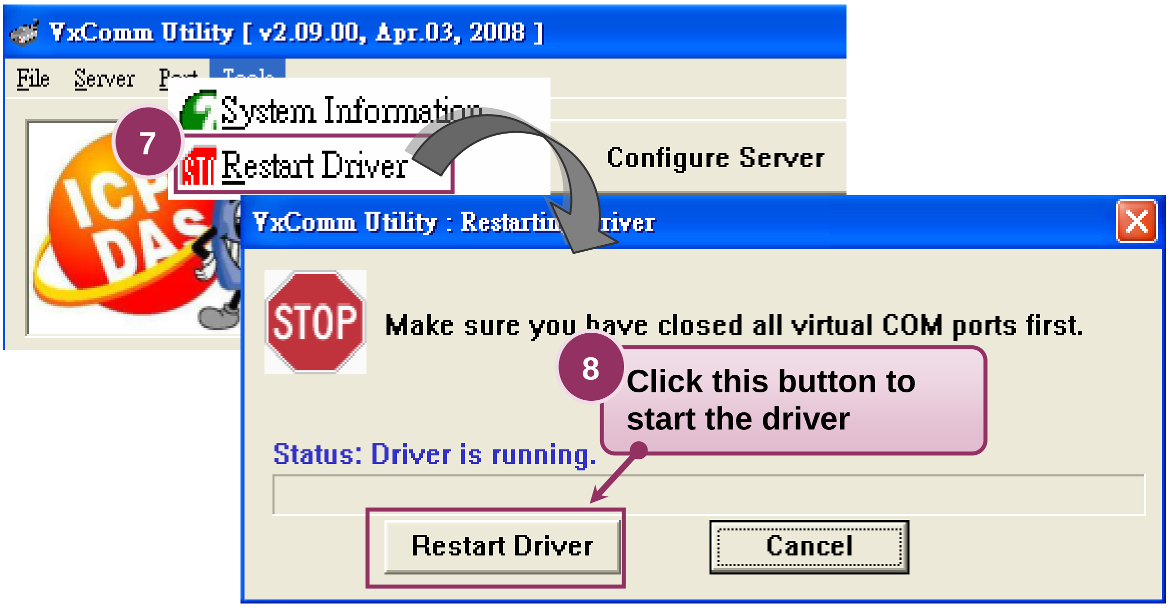
3. Check the Virtual COM port numbers on the PC.



Port	Virtual COM	Baudrate
Port I/O	Reserved	N/A
Port 1	COM2	Dynamic
Port 2	COM3	Dynamic
Port 3	COM4	Dynamic
Port 4	COM5	Dynamic
Port 5	COM6	Dynamic
Port 6	COM7	Dynamic
Port 7	COM8	Dynamic
Port 8	COM9	Dynamic

Name	Alias	IP Address	Sub-net Mask	Gateway	MAC Address
PDS-720	PDS-1F	10.0.8.243	255.255.255.0	10.0.8.254	00:0d:e0:d0:96:1c
PDS-782	N/A	10.0.8.20	255.255.255.0	10.0.8.254	00:0d:e0:20:00:07
ET-7050	N/A	10.0.8.115	255.255.255.0	10.0.8.254	00:0d:e0:d0:50:07

4. Click “**Restart Driver**” from the “**Tools**” menu, and then click the “**Restart Driver**” button to start the driver.


VxComm Utility [v2.09.00, Apr.03, 2008]
File Server Port Tools
System Information
7
Restart Driver
Configure Server
VxComm Utility : Restarting Driver
STOP
8
Click this button to start the driver
Restart Driver
Cancel

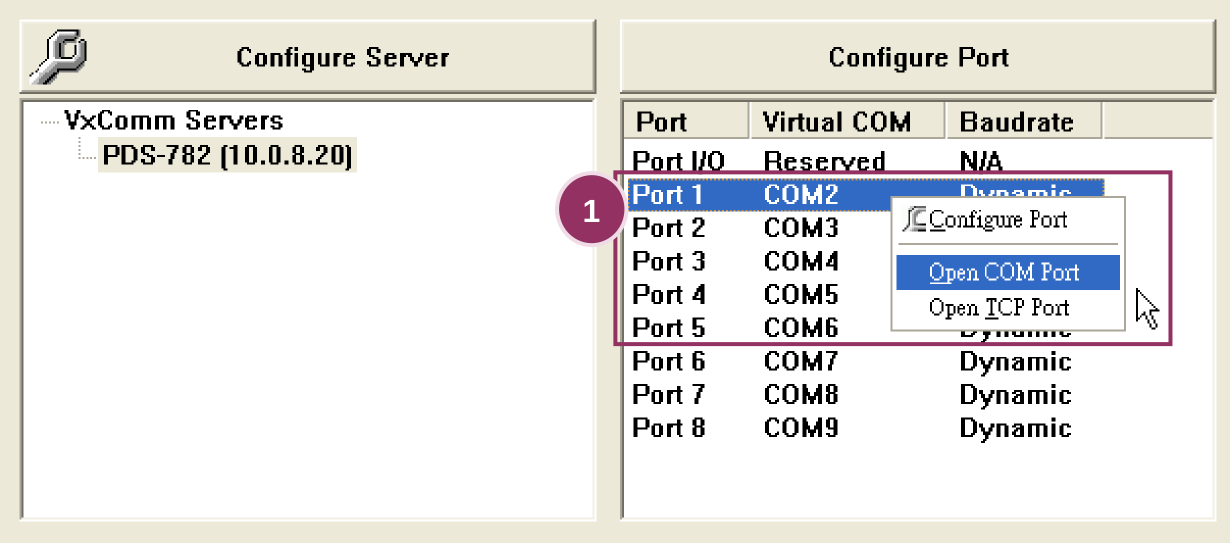
Make sure you have closed all virtual COM ports first.

Status: Driver is running.

Step4: Testing your PDS

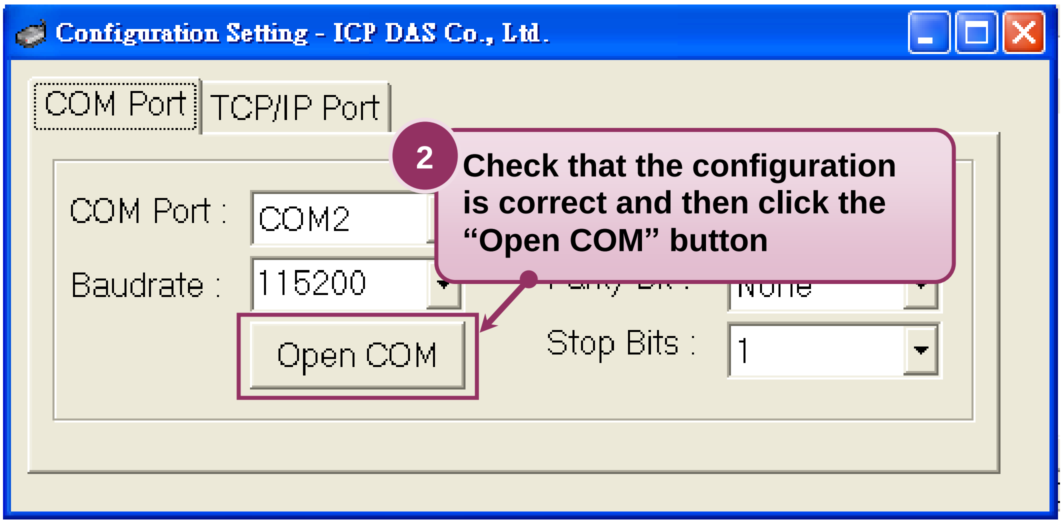
1. Connect the "**RxD1**" and the "**TxD1**" of the PDS module, as shown in the diagram in Step1.

2. Right click Port 1 and then choose the "**<u>Open COM Port</u>**" option.



Port	Virtual COM	Baudrate
Port I/O	Reserved	N/A
Port 1	COM2	Dynamic
Port 2	COM3	
Port 3	COM4	
Port 4	COM5	
Port 5	COM6	Dynamic
Port 6	COM7	Dynamic
Port 7	COM8	Dynamic
Port 8	COM9	Dynamic

1. Check that the configuration of the COM Port is correct and then click the "**<u>Open COM</u>**" button.



COM Port : COM2

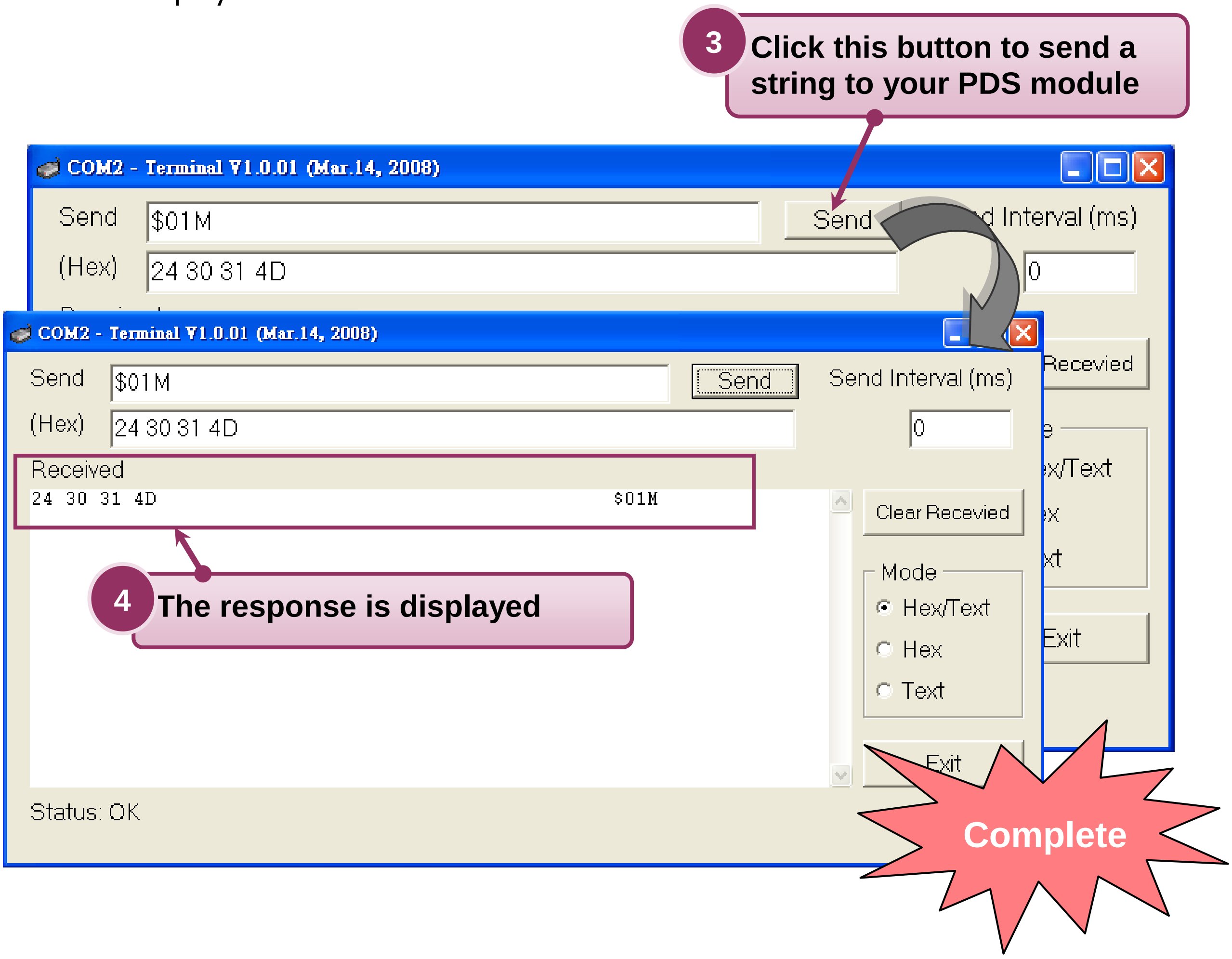
Baudrate : 115200

Parity Bit : None

Stop Bits : 1

Check that the configuration is correct and then click the "Open COM" button

2. Type a string in the send field then click the "**Send**" button. If a response is received, it will be displayed in the received field.



3 Click this button to send a string to your PDS module

COM2 - Terminal V1.0.01 (Mar.14, 2008)

Send: $01M

(Hex): 24 30 31 4D

Send Interval (ms): 0

Received

```
24  30  31  4D                                    $01M
```

4 The response is displayed

Mode: ● Hex/Text ○ Hex ○ Text

Status: OK

Complete

3. If the test is successful, the COM port program will be able to work directly by setting the Virtual COM Port number.

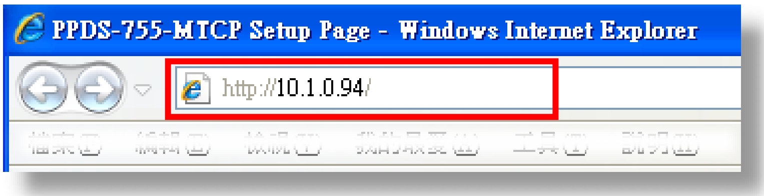
5. Configuration with Web Browser

Once the PDS series module has been correctly configured and is networking normally, the configuration details can be retrieved or amended using either the VxComm Utility or a standard web browser, such as IE, FireFox, or Mozilla, etc.

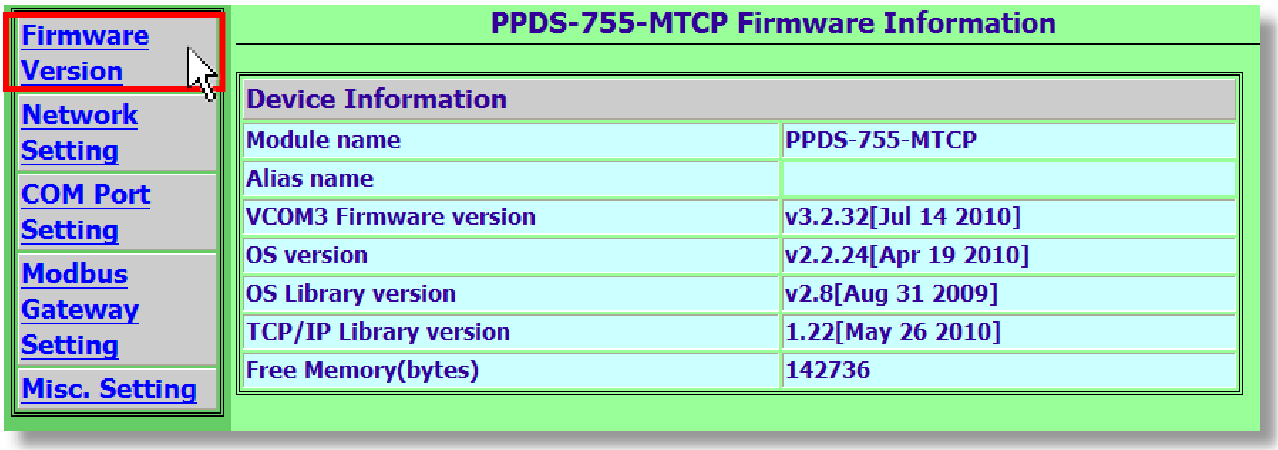
5.1 Connecting to the PDS Series Module

⚠ *If the COM port program is operating with a PDS module, changing the configuration will cause a program error.*

- Enter the IP address of the PDS series module in the Address field and press "**Enter**" to connect to the PDS series module.



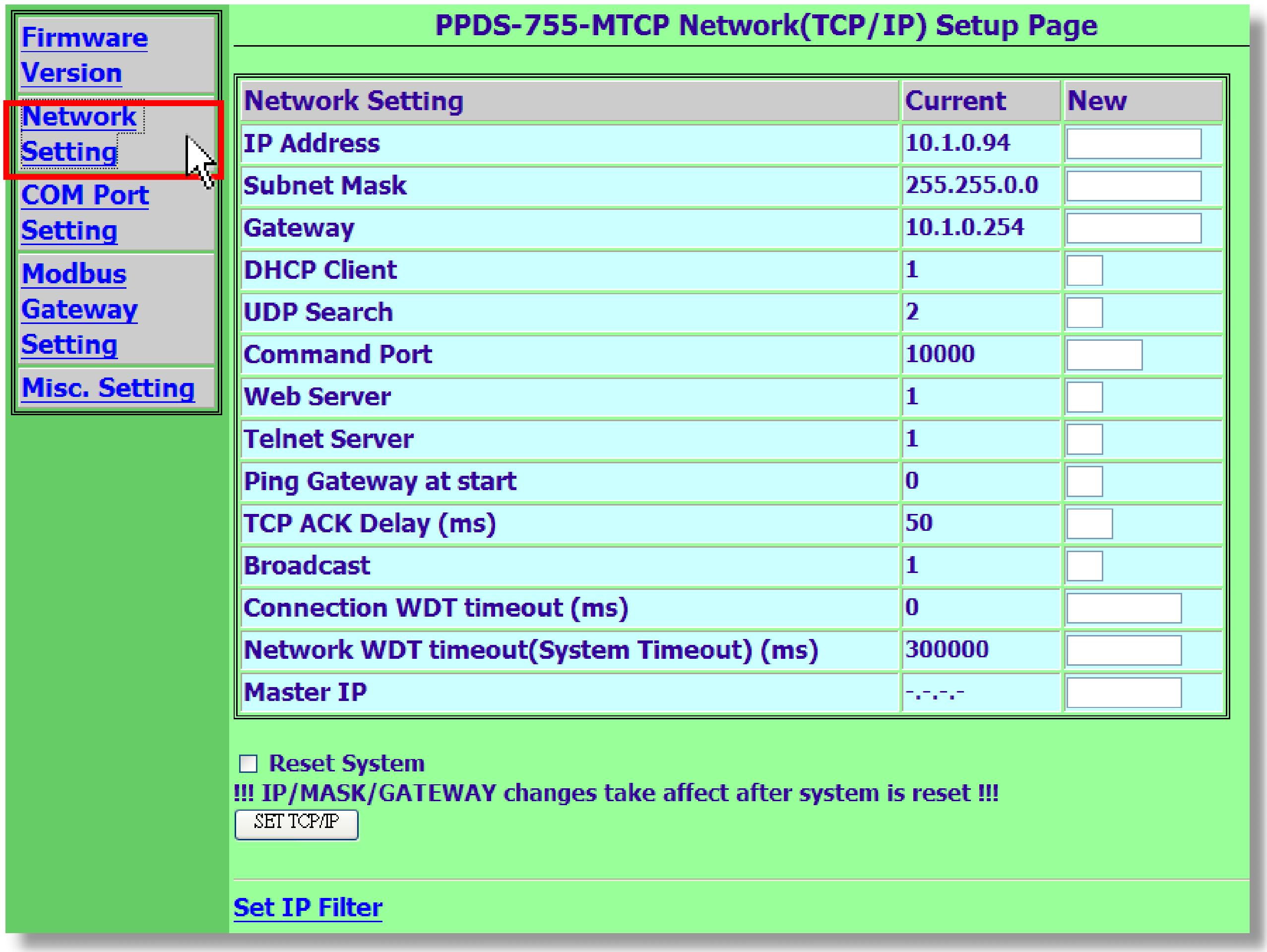
- When the browser connects to the PDS series module, the first page that will be display is the Firmware Information page.



PPDS-755-MTCP Firmware Information

Device Information	
Module name	PPDS-755-MTCP
Alias name	
VCOM3 Firmware version	v3.2.32[Jul 14 2010]
OS version	v2.2.24[Apr 19 2010]
OS Library version	v2.8[Aug 31 2009]
TCP/IP Library version	1.22[May 26 2010]
Free Memory(bytes)	142736

5.2 Network Settings



PPDS-755-MTCP Network(TCP/IP) Setup Page

Network Setting	Current	New
IP Address	10.1.0.94	
Subnet Mask	255.255.0.0	
Gateway	10.1.0.254	
DHCP Client	1	
UDP Search	2	
Command Port	10000	
Web Server	1	
Telnet Server	1	
Ping Gateway at start	0	
TCP ACK Delay (ms)	50	
Broadcast	1	
Connection WDT timeout (ms)	0	
Network WDT timeout(System Timeout) (ms)	300000	
Master IP	-.-.-.-	

□ Reset System

!!! IP/MASK/GATEWAY changes take affect after system is reset !!!

Network (TCP/IP) Setup page

- **IP Address**
- **Subnet Mask**
- **Gateway**

The above three items are the most important network settings and should always correspond to the LAN definition. If they do not match, the PDS series module will not operate correctly. If the settings are changed while the module is operating, any links to Virtual COM Port based applications currently in use will be lost and an error will occur.

- **DHCP Client** : 0 = disabled, 1 = enabled

It is recommended that the DHCP Client setting is kept as disabled, and using static network settings. This ensures your PDS series always using a fixed IP address, and you don’t need to configure the virtual COM mappings again and again.

- **UDP Search**: 0 = disabled, 1 = always enabled.

2 = enable the UDP Search function until another client is connected. (Default = 2)
By keeping the UDP search setting as 2, the PDS loading will be reduced. The VxComm Utility will not be able to search for this module until this module's clients are all disconnected.

- **Command Port:**

The default Command Port is 10000.

- **Web Server**
- **Telnet Server**

0 = disabled, 1 = enabled

- **Ping Gateway at start**: 0 = disabled, 1 = enabled.

If the setting is 1 (enabled), the PDS series module will send a ping packet to the gateway during the power-on stage. It is used to inform the gateway that a PDS (itself) has joined the network.

- **TCP ACK Delay (ms)**, default = 50.

PDS does not want to send an empty ACK followed by a TCP data packet 1ms later, every time. So it delays a little (TCP ACK Delay), and then can combine the ACK and data packet into one. This efficiency reduces the number of packets and reduces network loadings.

- **Broadcast**
  1 = receive UDP broadcast packets
  0 = reject UDP broadcast packets

- **Connection WDT timeout (ms)**: default = 0 (disabled), min. = 10000.

If the PDS series module does not receive any data from a client PC within the period of the "Connection WDT timeout", the module will close the connection to the client.

- **Network WDT timeout (ms)**: 0 = disabled, min. = 30000.

If the PDS series module does not receive any data from any of the clients within the period of the "Network WDT timeout", the module will reboot itself. The default setting should be 300000ms (= 300 seconds).

This setting is the same as "SystemTimeout" setting (unit: ms) on Console/Telnet command, and is the same as "/STxxx" in command line parameter (unit: seconds).

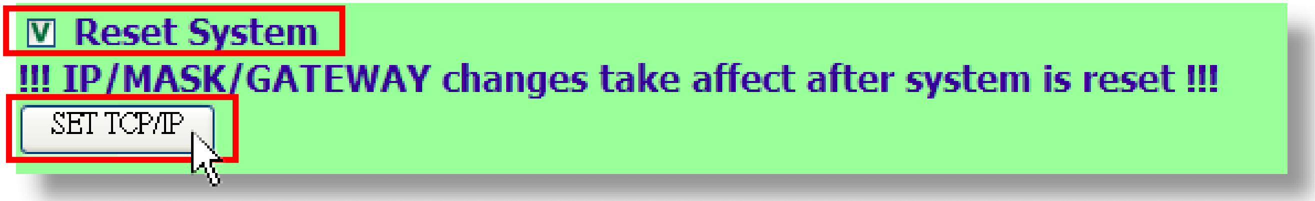
When user uses "config=RESET" Console/Telnet command to clear the EEPROM, the "Network WDT timeout" (SystemTimeout, /ST) setting will also be cleared to 0.

Users have to configure this setting again by "SystemTimeout" Console/Telnet command.

- **Master IP**: default = empty (disabled).

If the Master IP is set, only the client using Master IP can change the COM Port configuration. It is to prevent the COM Port configuration changed by other clients.

After setting the new configuration, click the "**Set TCP/IP**" button to save the new settings to the PDS series module. If the "**Reset System**" option is checked, the PDS series module will reboot itself after the saving operation is complete, otherwise the original settings will still be valid until the next power-on.

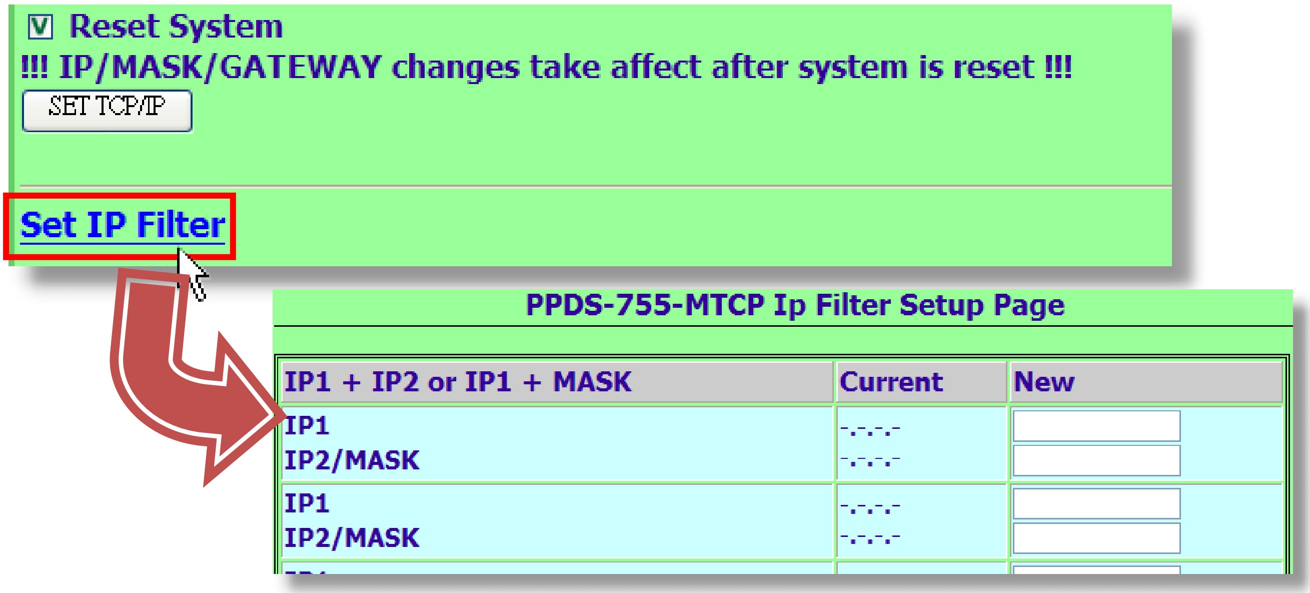


☑ Reset System

!!! IP/MASK/GATEWAY changes take affect after system is reset !!!

5.3 IP filter setting

The IP filter setting limits which client PCs are able to link to the PDS series module via specific IP addresses. When one or more IP addresses are set in the filter table, only client PCs where the IP address is included in the range listed of the filter table will be able to connect to the PDS series module. Any requests from other PCs will be rejected



☑ Reset System

!!! IP/MASK/GATEWAY changes take affect after system is reset !!!

PPDS-755-MTCP Ip Filter Setup Page

IP1 + IP2 or IP1 + MASK	Current	New
IP1 IP2/MASK	-.-.-.- -.-.-.-	
IP1 IP2/MASK	-.-.-.- -.-.-.-	

- **Set IP1 only**: only clients who's IP address is included in the filter table are able to connect to the PDS series module.

- **Set IP1 + IP2**: set a range of IP address as a starting and ending point. The setting allows clients who's IP address is included in the range are able to connect to the PDS series module.

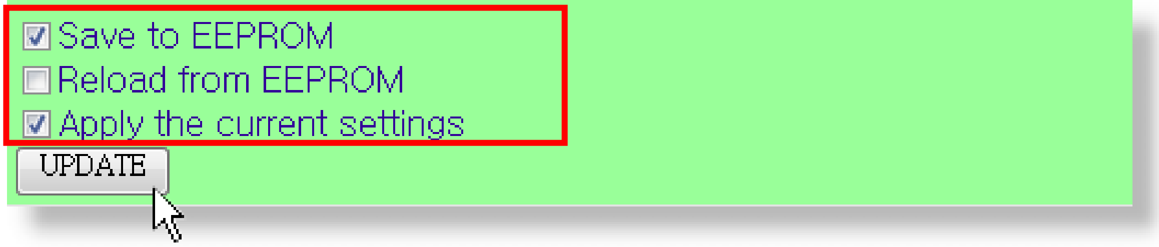
- **Set IP1+Mask**: set the IP filter range as:
  (IP1 & Mask) + 0 ~ (IP1 & Mask) + (~Mask).

Only clients who's IP address is included in the range are able to connect to the PDS series module. For instance:

IP1 = 10.0.9.5, mask = 255.255.255.0
IP1 & MASK = 10.0.9.0, ~mask = 0.0.0.255

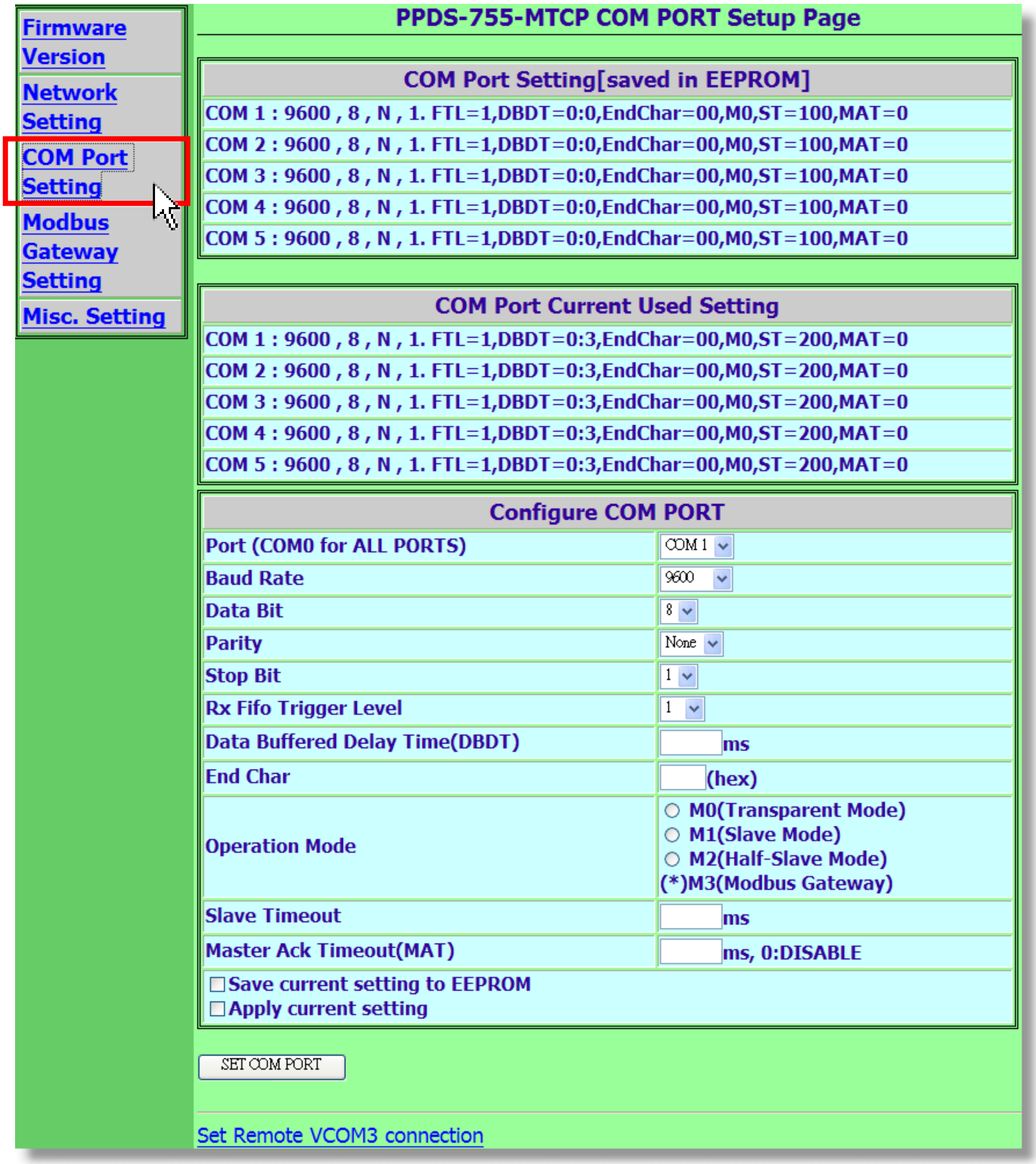
This allows clients who's IP address is included in the range of 10.0.9.0 ~ 10.0.9.255 are able to connect to the PDS series module.

- Check the "**Save to EEPROM**" and "**Apply the current settings**", and then click the "**Update**" button to validate the settings.



☑ Save to EEPROM
□ Reload from EEPROM
☑ Apply the current settings

5.4 COM Port Settings



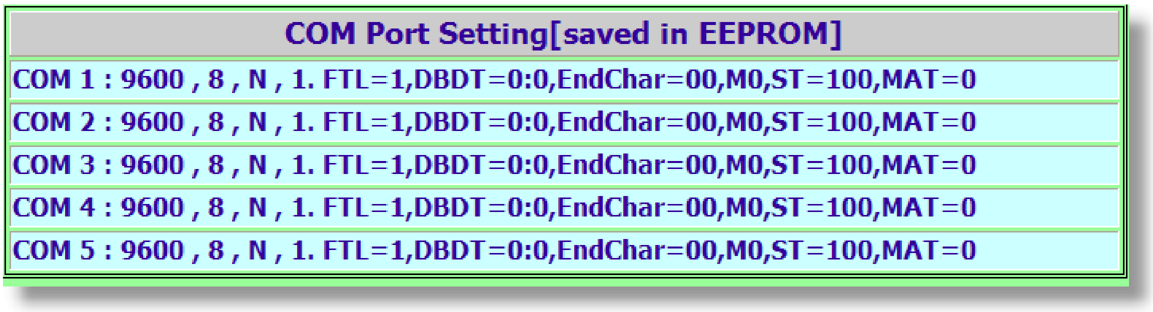
PPDS-755-MTCP COM PORT Setup Page

COM Port Setting[saved in EEPROM]
COM 1 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0
COM 2 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0
COM 3 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0
COM 4 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0
COM 5 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0

COM Port Current Used Setting
COM 1 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0
COM 2 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0
COM 3 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0
COM 4 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0
COM 5 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0

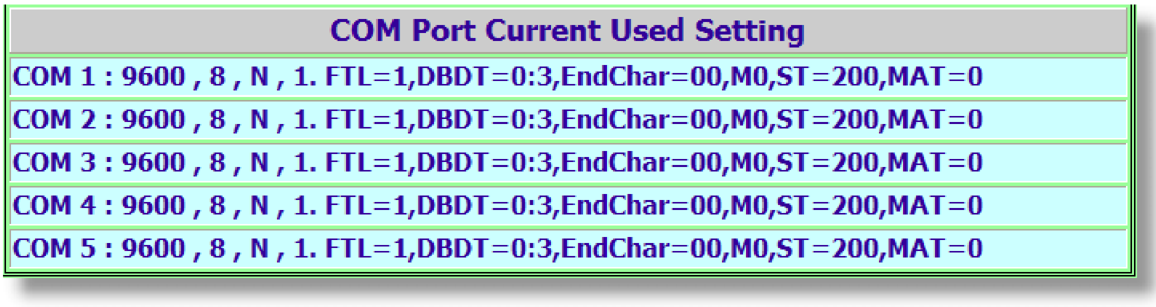
Configure COM PORT	
Port (COM0 for ALL PORTS)	COM 1
Baud Rate	9600
Data Bit	8
Parity	None
Stop Bit	1
Rx Fifo Trigger Level	1
Data Buffered Delay Time(DBDT)	ms
End Char	(hex)
Operation Mode	○ M0(Transparent Mode) ○ M1(Slave Mode) ○ M2(Half-Slave Mode) (*)M3(Modbus Gateway)
Slave Timeout	ms
Master Ack Timeout(MAT)	ms, 0:DISABLE
□ Save current setting to EEPROM □ Apply current setting	

- The COM Port Settings list is saved in the EEPROM on the PDS series module.



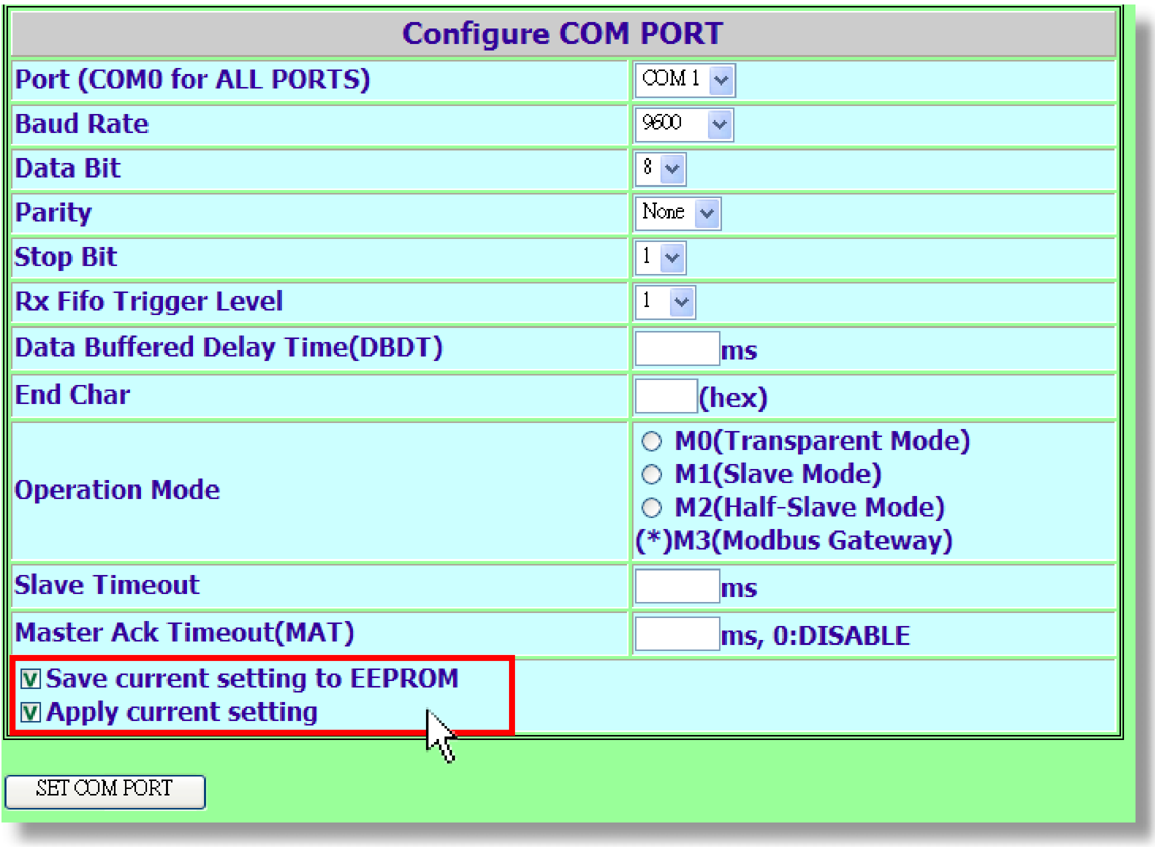
COM Port Setting[saved in EEPROM]
COM 1 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0
COM 2 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0
COM 3 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0
COM 4 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0
COM 5 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0

- The Currently Used COM Port Settings list.



COM Port Current Used Setting
COM 1 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0
COM 2 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0
COM 3 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0
COM 4 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0
COM 5 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0

- The COM Port Settings area



Configure COM PORT	
Port (COM0 for ALL PORTS)	COM 1
Baud Rate	9600
Data Bit	8
Parity	None
Stop Bit	1
Rx Fifo Trigger Level	1
Data Buffered Delay Time(DBDT)	ms
End Char	(hex)
Operation Mode	○ M0(Transparent Mode) ○ M1(Slave Mode) ○ M2(Half-Slave Mode) (*)M3(Modbus Gateway)
Slave Timeout	ms
Master Ack Timeout(MAT)	ms, 0:DISABLE
☑ Save current setting to EEPROM ☑ Apply current setting	

- **Save current settings to EEPROM:** If the "**Set COM Port**" button is clicked without checking "**Save current settings to EEPROM**", option the new settings will be saved to the PDS-220Fx series only and the new settings will be valid after the next power-on.

- **Apply current settings:** If the "**Apply current setting**" checked when the "**Set COM Port**" button is clicked, the new settings will be valid immediately.

.

Note: After configuring COM Port settings, please confirm that the "Save current settings to EEPROM" and "Apply Current settings" items are checked, and then click the "SET COM PORT" button.

- **Port**: The COM Port number on the PDS series module.
- **Baud Rate, Data Bits, Parity**
- **Stops Bits, End Character**:

  The configuration settings should match the serial device used.

- **Fifo Trig. Level**: FIFO trigger level

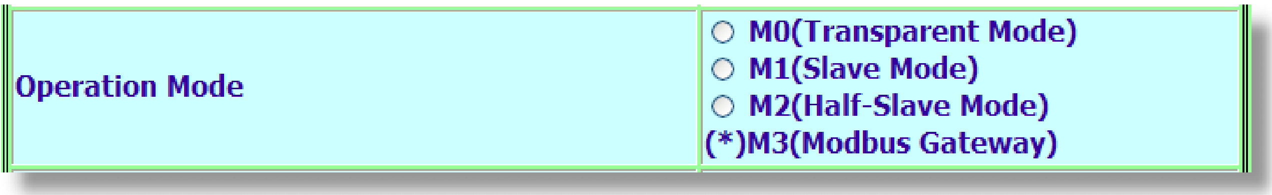
This option is used to set the number of characters that the COM Port can receive at once time, the PDS will move the data from the COM Port FIFO to the PDS. If the amount of data transferred is large and uses a transfer speed (115200 bps), setting a smaller value is helpful in preventing data loss.

- **DBDT (ms)**: Data buffer delay timeout

When the COM port does not receive data from devices connected over the period of DBDT setting, the PDS will determine that the data transfer is over and return to process next tasks.

- **Operation Mode**:

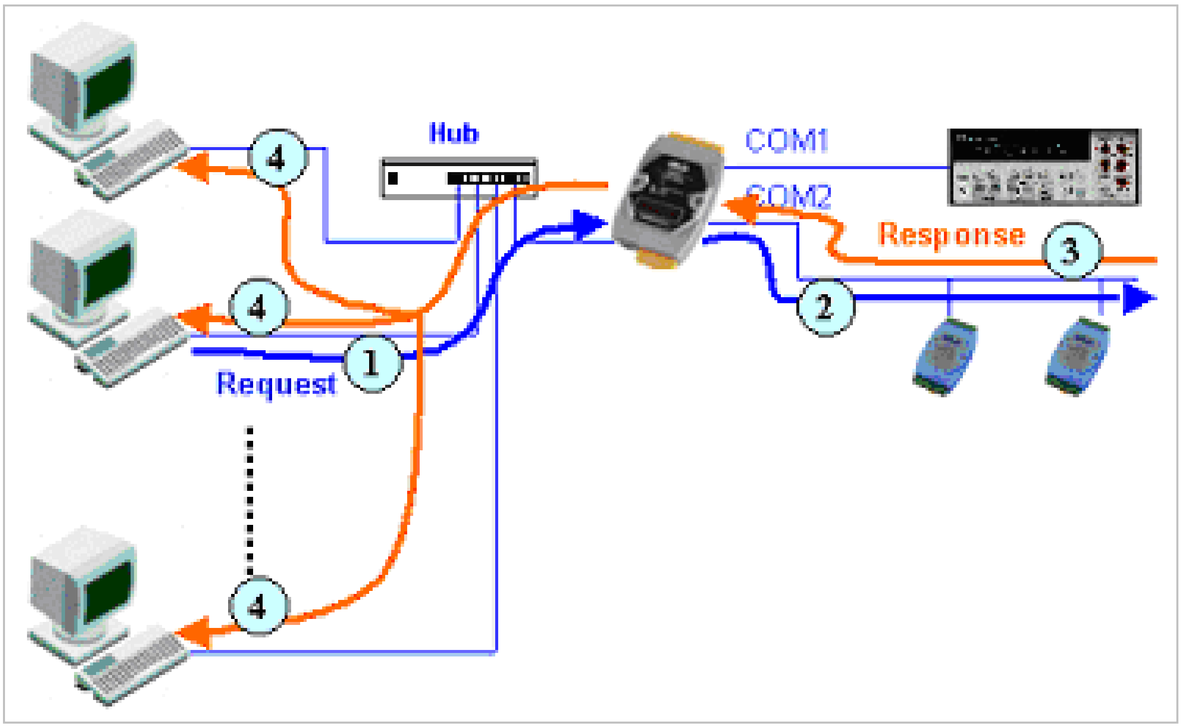
  M0 (Transparent Mode),
  M1 (Slave Mode),
  M2 (Half-Slave Mode),
  M3 (Modbus Gateway)



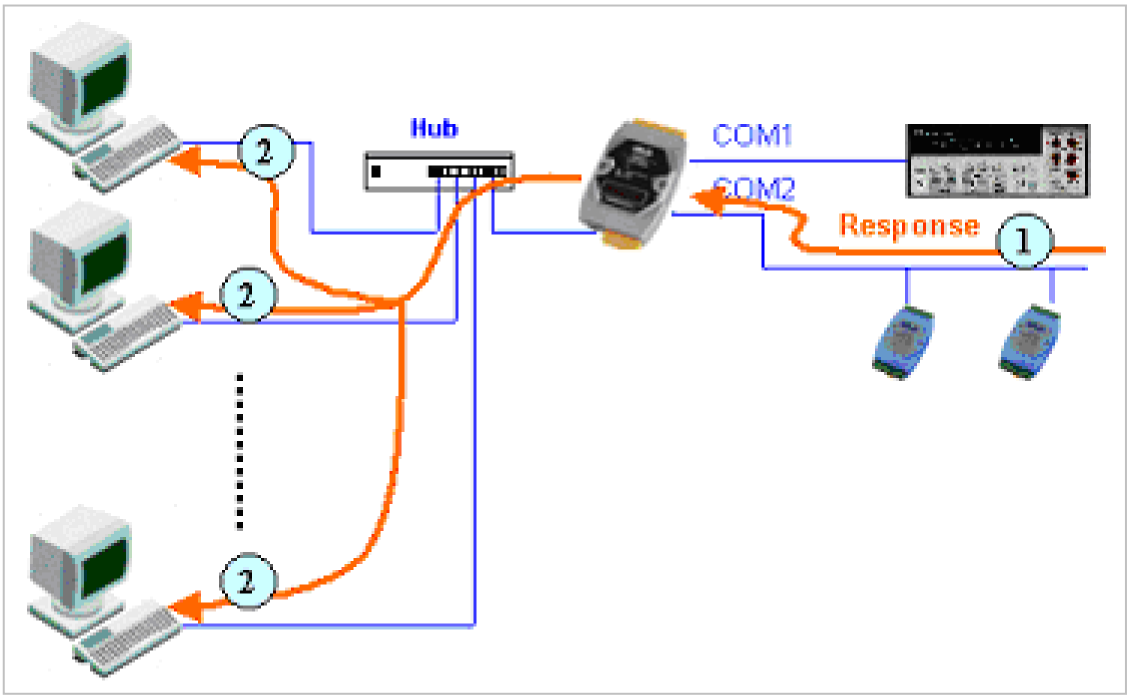
Operation Mode	○ M0(Transparent Mode) ○ M1(Slave Mode) ○ M2(Half-Slave Mode) (*)M3(Modbus Gateway)

M0: Transparent Mode (Multi-echo mode)

Condition 1: One client sends a request to the PDS series module to access each device. The PDS series module echoes the data from each device to each connected client.

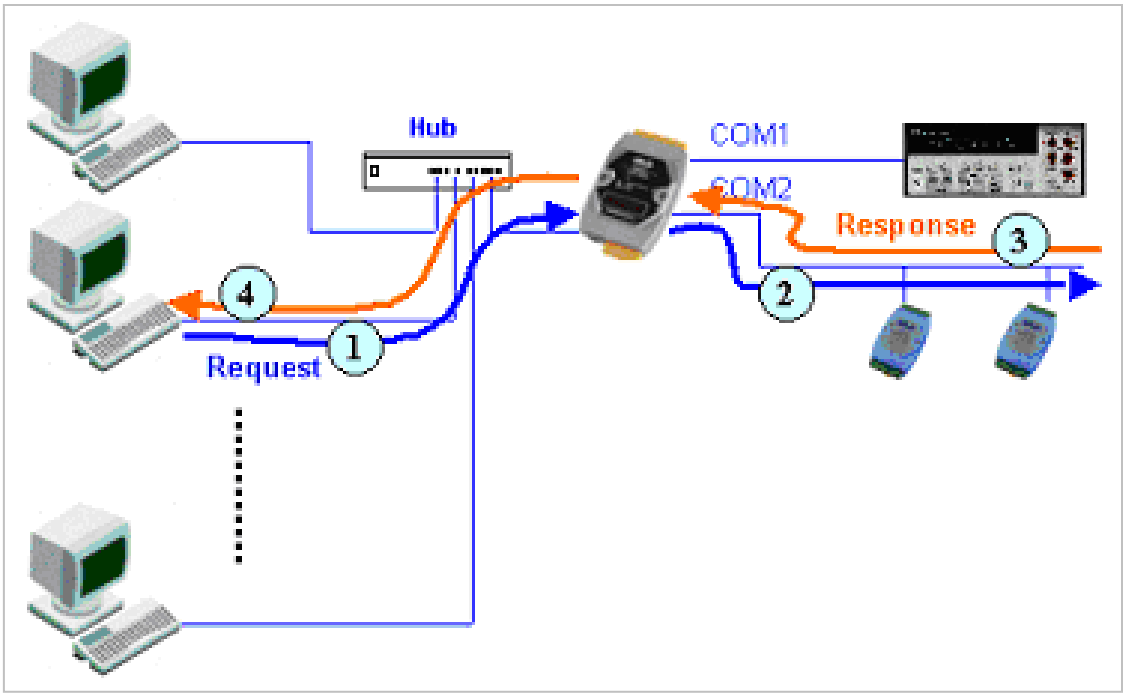


Condition 2: No clients send any requests to the PDS series module. The PDS series module echoes data from the devices to each connected client.

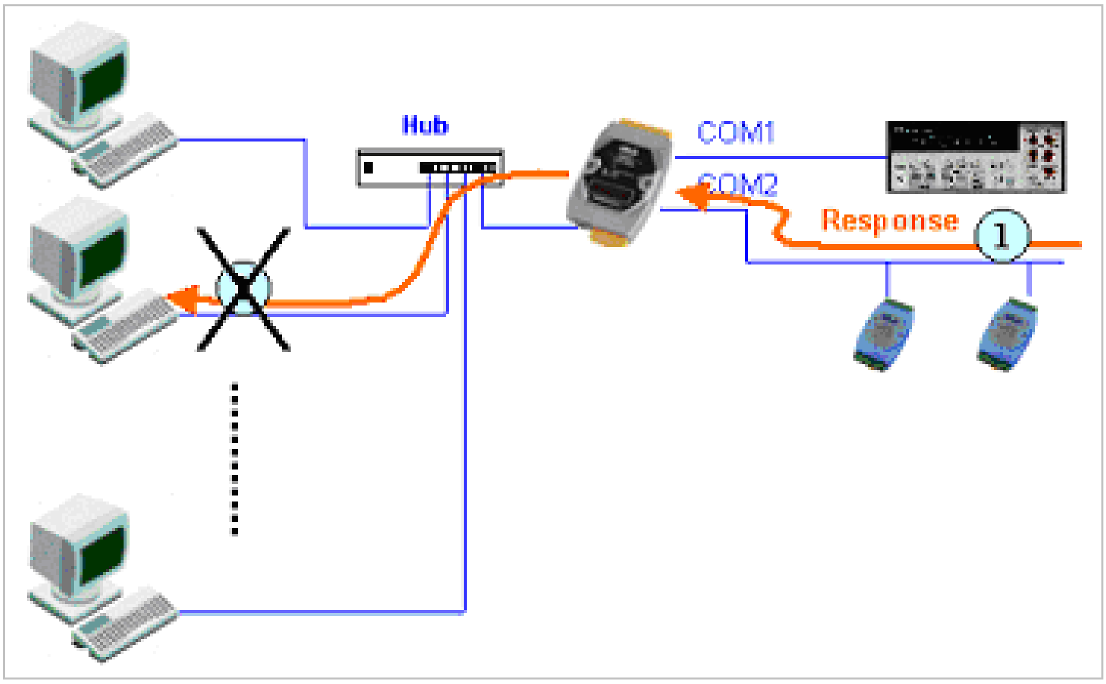


■ M1: Slave Mode (Single-echo mode)

Condition 1: One client sends a request to the PDS series module to access the other devices. The PDS series module echoes data from the devices to the client that requested the service.



Condition 2: No clients send any requests to the PDS series module. The PDS series module doesn't echo any data from the devices to any client.



- **M2: Half-Slave Mode**

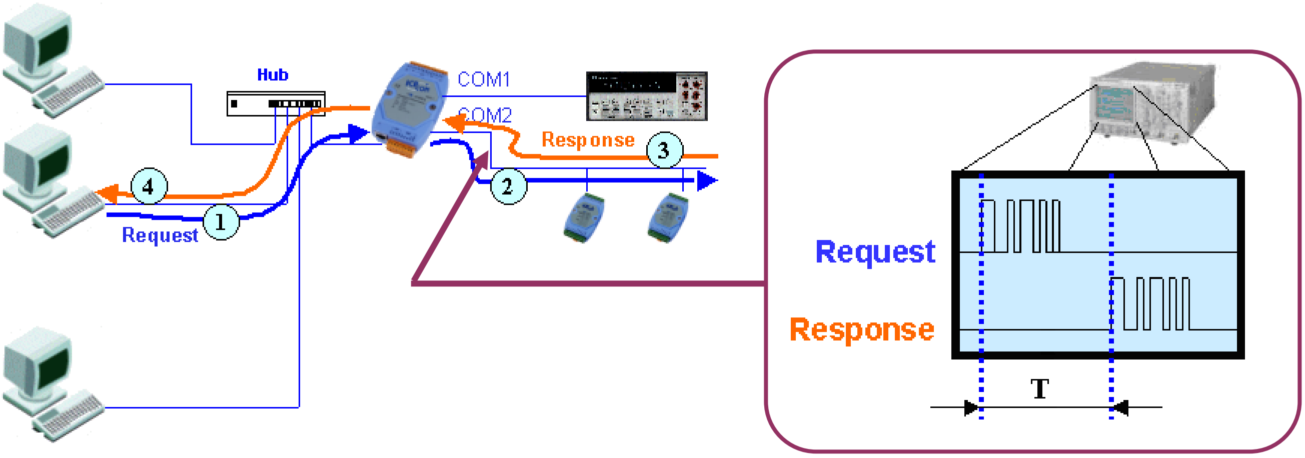
If only one client connects to the PDS series module, the operation mode same as M0 mode, please refer to page 75 (**M0: Transparent Mode**); and if there are two or more clients connecting to the PDS series module, the operation mode same as M1 mode please refer to page 76 (**M1: Slave Mode**).

- **M3: Modbus Gateway**

The more detailed information regarding Modbus Gateway applications settings, please refer to Section 6 "Modbus Testing and Protocol".

Slave Timeout (ms):

In M1, the slave mode timeout setting is use to set the waiting time after last character of the request sent to the device. If the device does not respond within the timeout value, the PDS series module will return a timeout error and process next request.

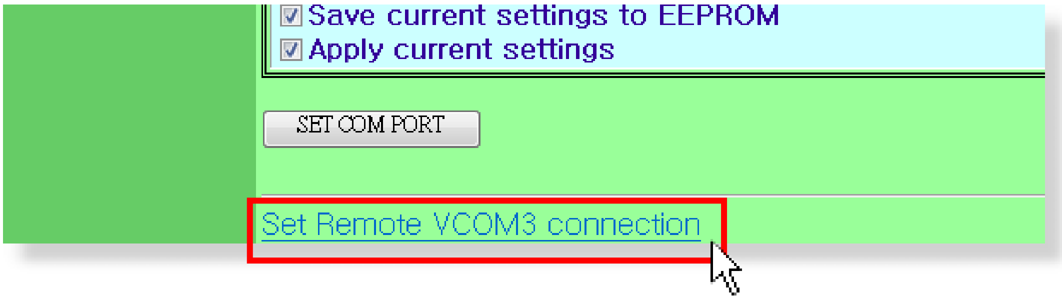


Master Ack Timeout (MAT): (ms) (0: DISABLE)

If the PDS series module does not receive any data from Slave for a certain period, the PDS series module will return an ACK character to Master and process next request based on the configured timeout value.

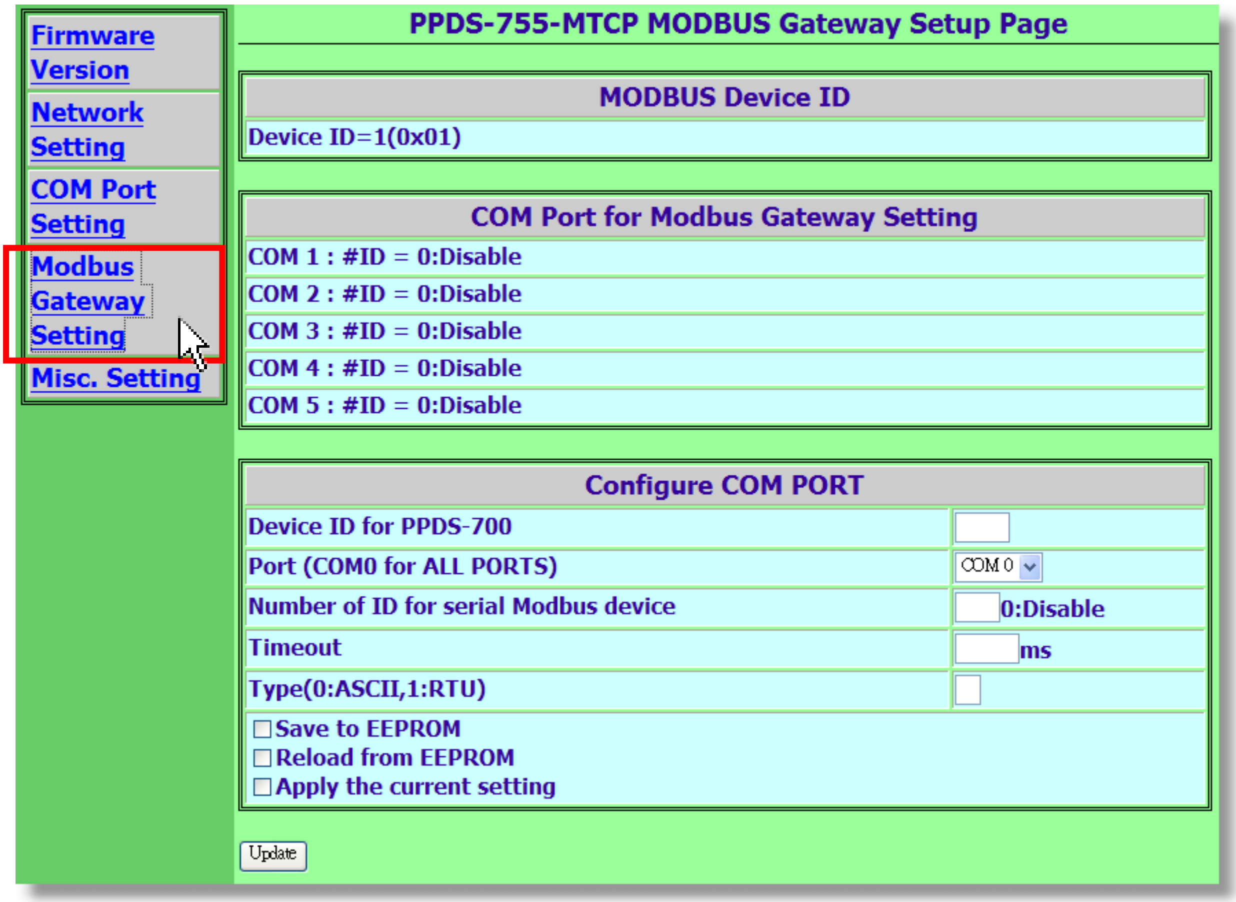
Set Remote VCOM3 connection:

Sets the pair-connection functions, the more detailed information regarding pair-connection applications settings, please refer to the Section 5.7 Pair Connection Settings.



☑ Save current settings to EEPROM
☑ Apply current settings
SET COM PORT
Set Remote VCOM3 connection

5.5 Modbus Gateway Setting



PPDS-755-MTCP MODBUS Gateway Setup Page

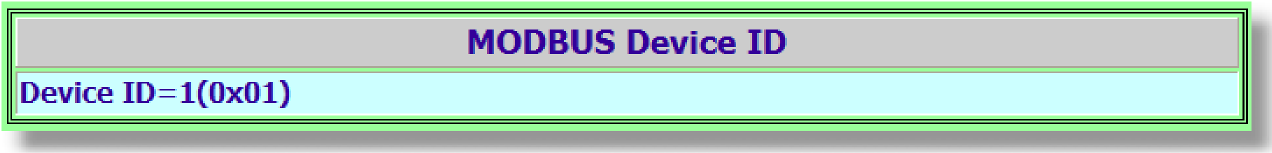
MODBUS Device ID
Device ID=1(0x01)

COM Port for Modbus Gateway Setting
COM 1 : #ID = 0:Disable
COM 2 : #ID = 0:Disable
COM 3 : #ID = 0:Disable
COM 4 : #ID = 0:Disable
COM 5 : #ID = 0:Disable

Configure COM PORT	
Device ID for PPDS-700	
Port (COM0 for ALL PORTS)	COM 0
Number of ID for serial Modbus device	0:Disable
Timeout	ms
Type(0:ASCII,1:RTU)	

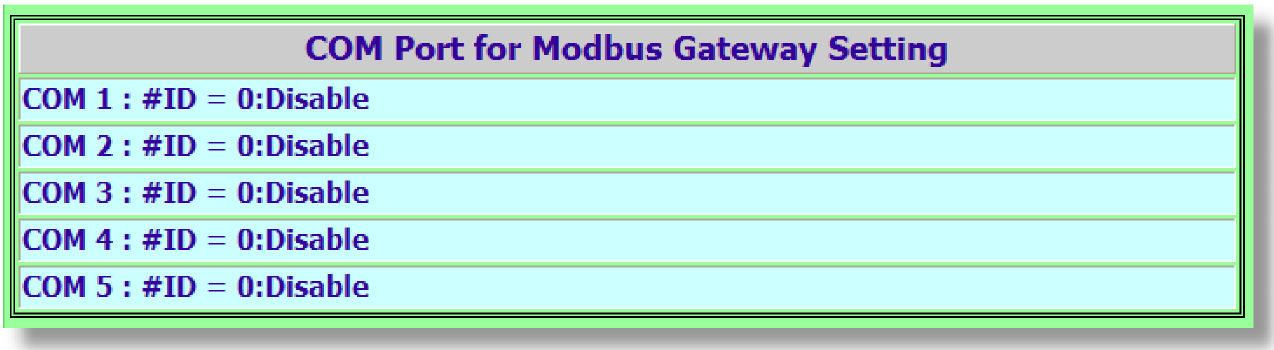
□ Save to EEPROM
□ Reload from EEPROM
□ Apply the current setting

- Check the Modbus Device ID of PPDS series module:



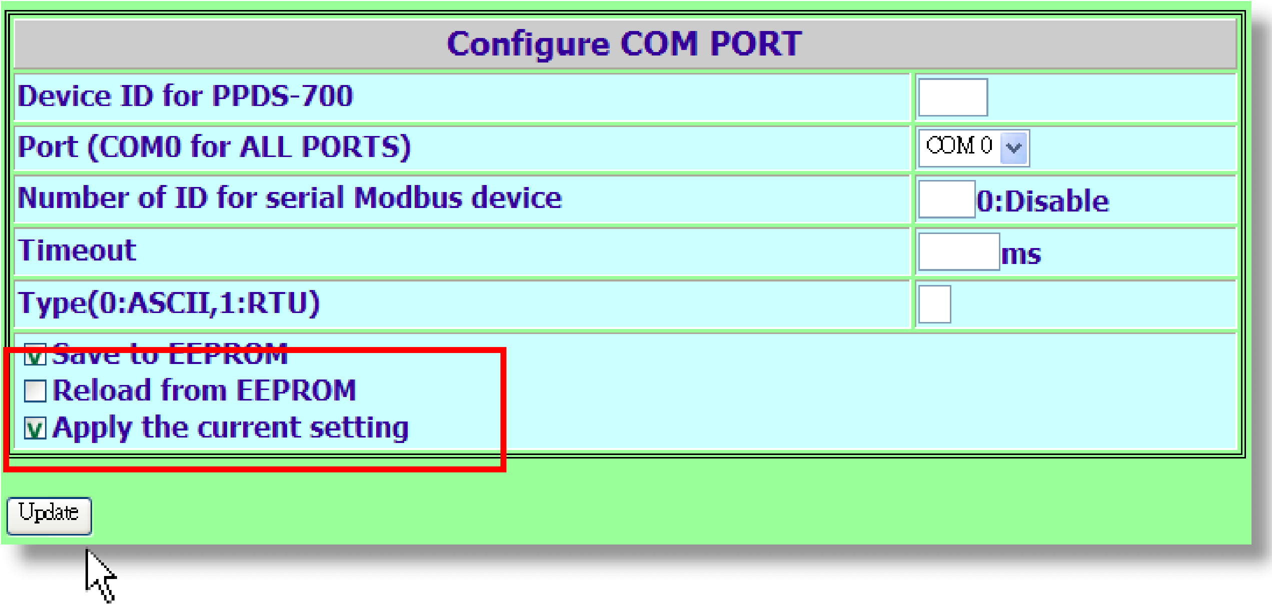
MODBUS Device ID
Device ID=1(0x01)

- Check the COM Port for Modbus Gateway settings:



COM Port for Modbus Gateway Setting
COM 1 : #ID = 0:Disable
COM 2 : #ID = 0:Disable
COM 3 : #ID = 0:Disable
COM 4 : #ID = 0:Disable
COM 5 : #ID = 0:Disable

- The COM Port Settings area:



Configure COM PORT	
Device ID for PPDS-700	
Port (COM0 for ALL PORTS)	COM 0
Number of ID for serial Modbus device	0:Disable
Timeout	ms
Type(0:ASCII,1:RTU)	

☑ Save to EEPROM

☐ Reload from EEPROM

☑ Apply the current setting

- **Device ID**: Sets the device ID of PPDS series module.
- **Port (COM0 for ALL Ports):** Select the COM port by you used for PPDS series module.

- **Number of ID for serial Modbus device**: Sets the device ID of serial Modbus device. (0: Disable)

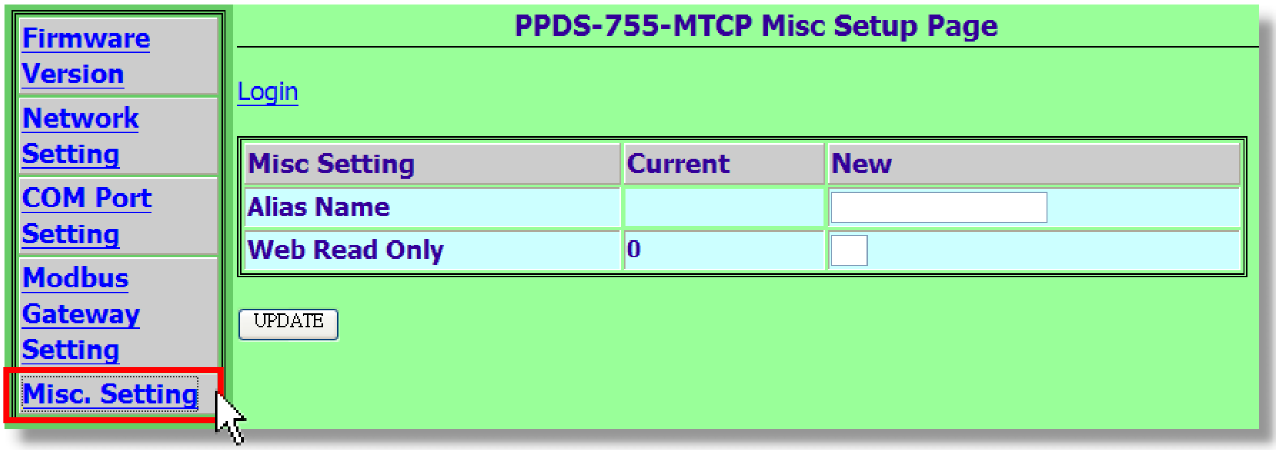
- **Timeout**: If the PPDS series module does not receive any data from RTU Slave for a certain period, the PPDS series module will return a timeout error to PC based on the configured timeout value.

- **Type (0: ASCII, 1:RTU):** Sets the Modbus type. (Modbus ASCII or Modbus RTU)

The more detailed information regarding Modbus Gateway applications settings, please refer to Section 6 "Modbus Testing and Protocol".

Note: After configuring COM Port settings, please confirm that the "Save current settings to EEPROM" and "Apply Current settings" items are checked, and then click the "Update" button.

5.6 Miscellaneous Setting



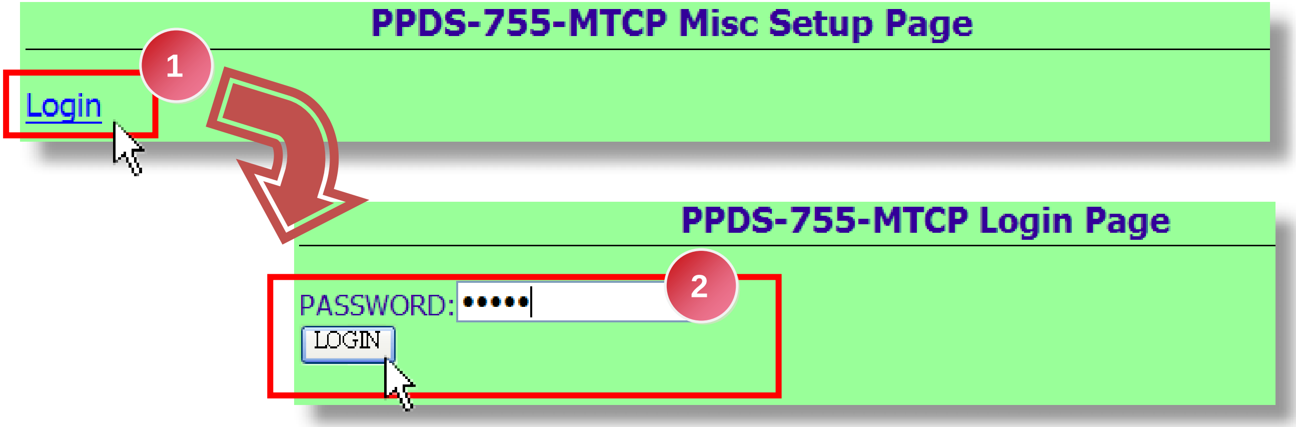
Misc Setting	Current	New
Alias Name		
Web Read Only	0	□

- **Alias Name**: allocates an alias to the PDS series module.
- **Web Read Only**: 0 = disabled, 1 = enabled

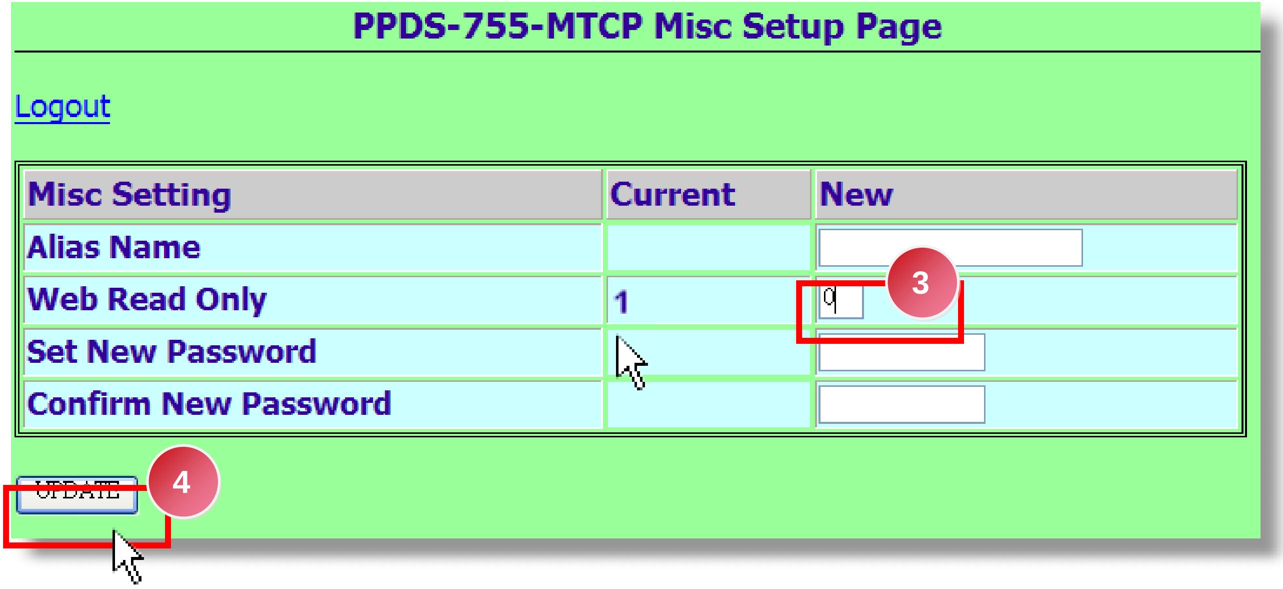
> ⚠ *If the "**Web Read Only**" properly is set to 1, enabled, the web server will **not** be able to save any new configurations to the PDS series module. To disable the "**Web Read Only**" property, refer to the information below.*

- **Login**: used to disable the "**Web Read Only**" property or to set a new password.

1. Enter the password (default is **admin**) and click the "**LOGIN**" button to proceed to the settings page.



2. Set the new “**Web Read Only**” properly = 0 and click the “**UPDATE**” button.

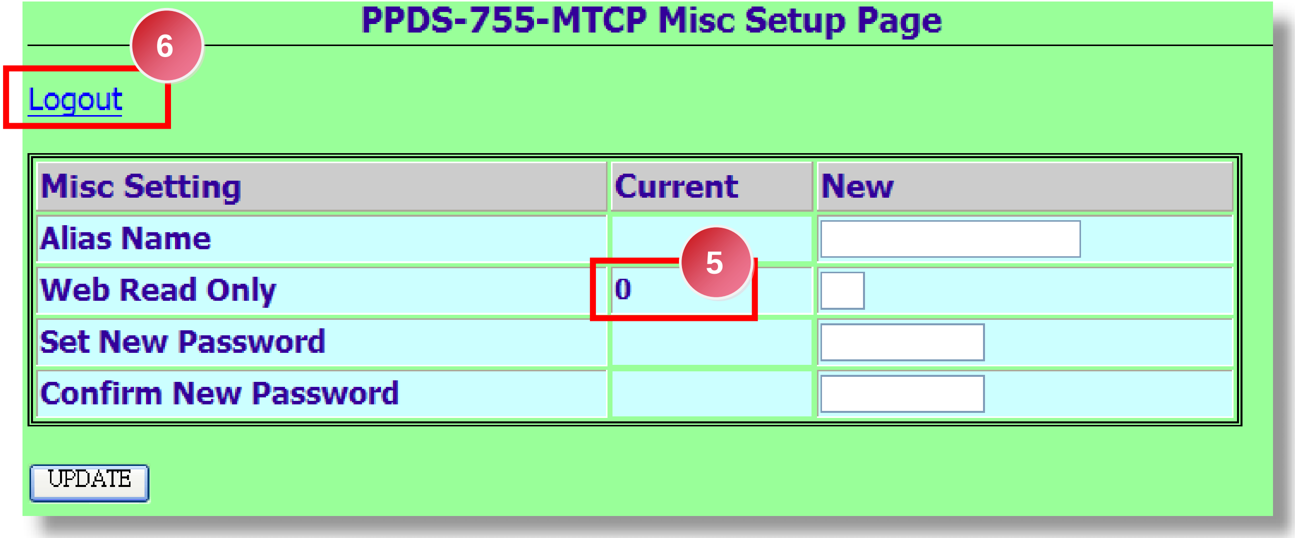


PPDS-755-MTCP Misc Setup Page

Logout

Misc Setting	Current	New
Alias Name		
Web Read Only	1	0
Set New Password		
Confirm New Password		

3. Check that the current the “**Web Read Only**” = 0 and then click “**Logout**” to complete the operation.



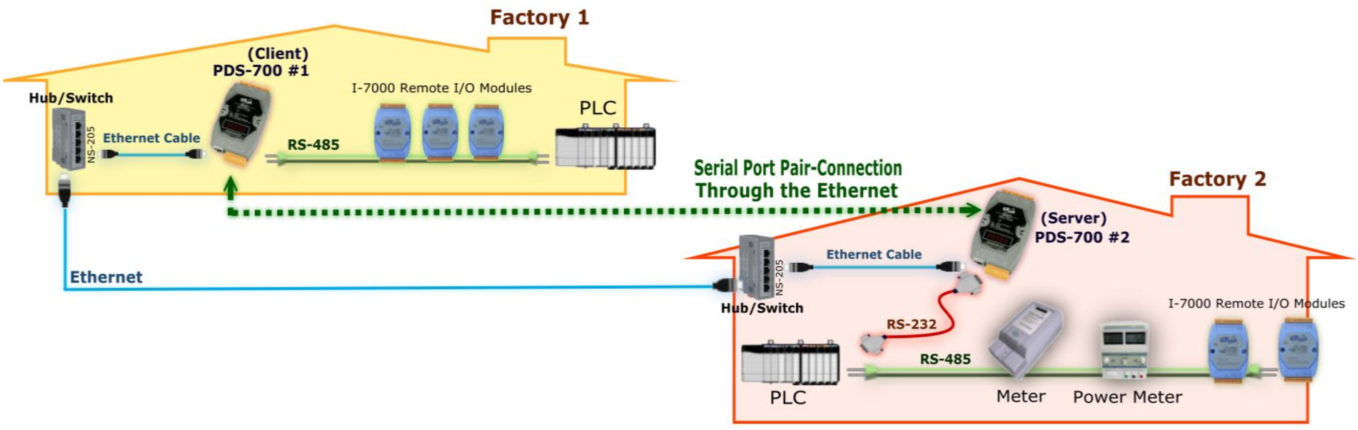
PPDS-755-MTCP Misc Setup Page

Logout

Misc Setting	Current	New
Alias Name		
Web Read Only	0	
Set New Password		
Confirm New Password		

4. User can restore PDS password to default value “**admin**” by using “config=RESET” console command (refer to section Console/Telnet Commands List). This command sets most configurations of PDS to factory setting. It requires rebooting the PDS for loading new configuration (includes default password).

5.7 Pair-Connection Setting

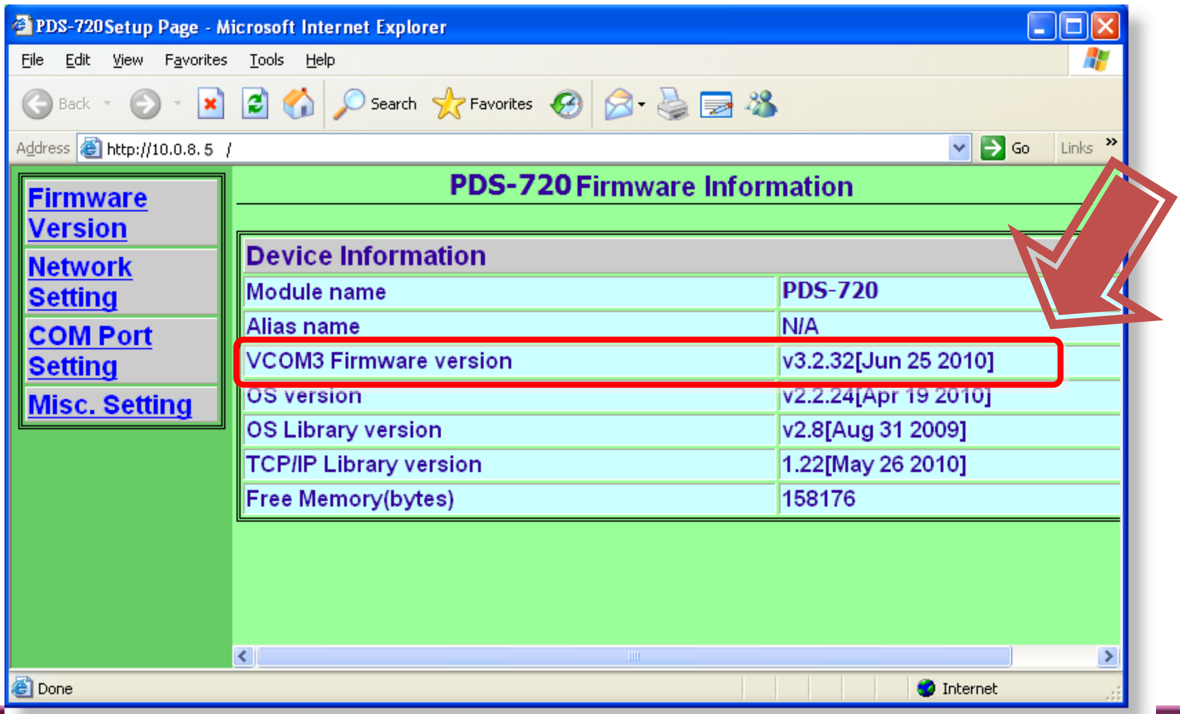


For example:

Item	Model Name	Server mode	Configuration IP Address
PDS-700 #1	PDS-720	Client	10.0.8.5
PDS-700 #2	PDS-720	Server	10.0.8.6

1. Contact your Network Administrator to obtain a correct and workable network configuration (such as IP/Mask/Gateway) for PDS series modules. Please also refer to section "“4 Setting up the PDS module”.

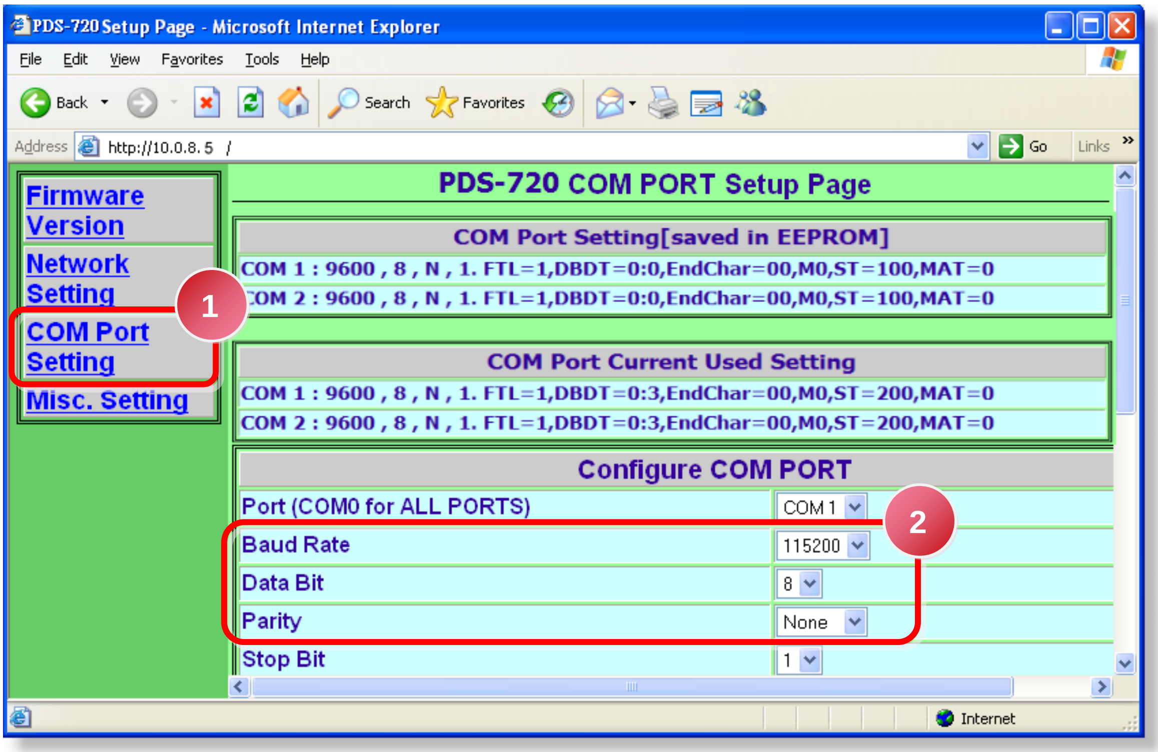
2. Enter the configuration of PDS-700#1(Client) web server and then check your firmware version is **v3.2.31[Jun 19 2009] or later.**



PDS-720 Firmware Information

Device Information	
Module name	PDS-720
Alias name	N/A
VCOM3 Firmware version	v3.2.32[Jun 25 2010]
OS version	v2.2.24[Apr 19 2010]
OS Library version	v2.8[Aug 31 2009]
TCP/IP Library version	1.22[May 26 2010]
Free Memory(bytes)	158176

3. Click the "**COM Port Setting**" tag to enter the settings page, and select the "**115200**" in Baud Rate field and "**8, None, 1**" in data format field for example.


PDS-720 Setup Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://10.0.8.5 / Go Links
Firmware Version
Network Setting
COM Port Setting
Misc. Setting
1
2
Internet

PDS-720 COM PORT Setup Page

COM Port Setting[saved in EEPROM]

COM 1 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0
COM 2 : 9600 , 8 , N , 1. FTL=1,DBDT=0:0,EndChar=00,M0,ST=100,MAT=0

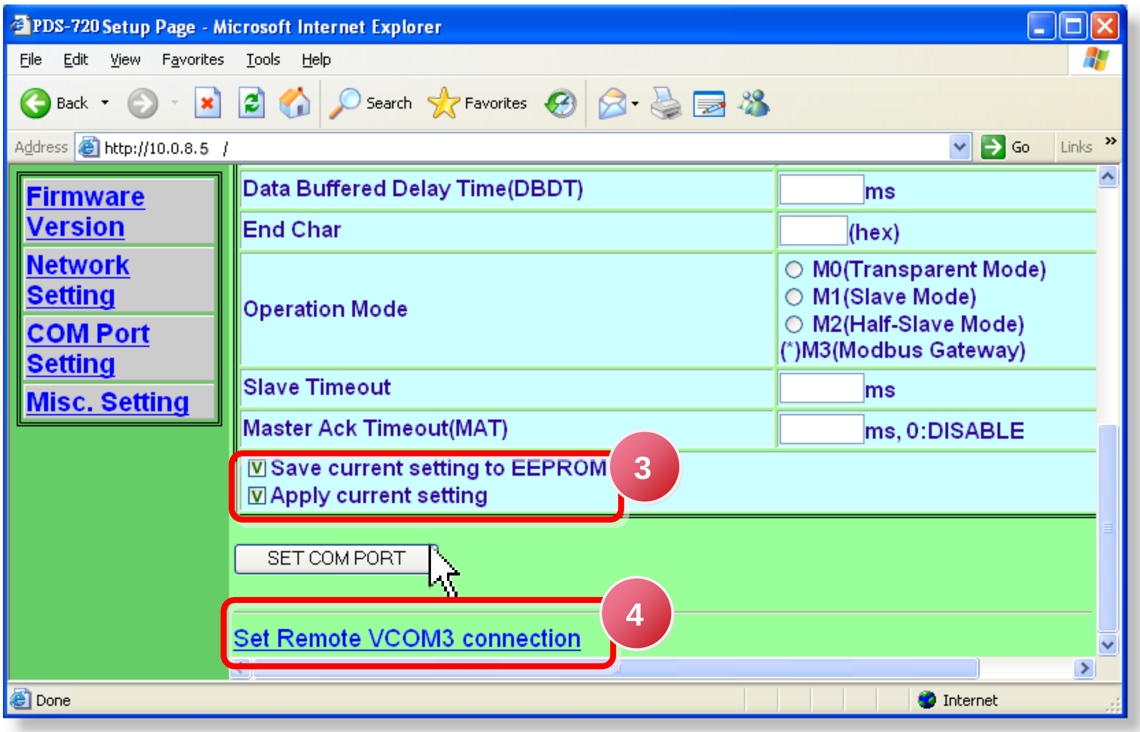
COM Port Current Used Setting

COM 1 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0
COM 2 : 9600 , 8 , N , 1. FTL=1,DBDT=0:3,EndChar=00,M0,ST=200,MAT=0

Configure COM PORT

Port (COM0 for ALL PORTS)	COM 1
Baud Rate	115200
Data Bit	8
Parity	None
Stop Bit	1

4. Check the "**Save current settings to EEPROM**" and "**Apply Current settings**" items, and then click the "**SET COM PORT**" button to finish configuration.
5. Click "**Set Remote VCOM3 connection**" at the PDS-700#1 (Client) COM PORT setup page.

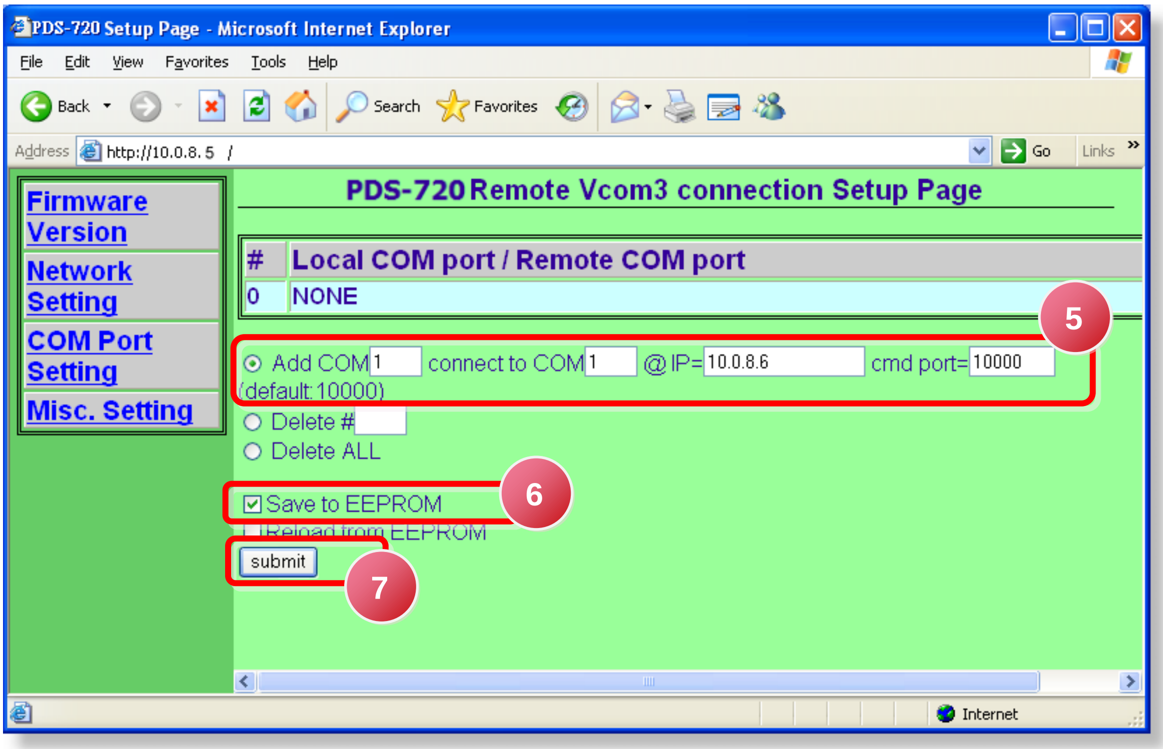

PDS-720 Setup Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://10.0.8.5 / Go Links
Firmware Version
Network Setting
COM Port Setting
Misc. Setting
3
4
SET COM PORT
Done
Internet

Data Buffered Delay Time(DBDT)	ms
End Char	(hex)
Operation Mode	○ M0(Transparent Mode) ○ M1(Slave Mode) ○ M2(Half-Slave Mode) (*)M3(Modbus Gateway)
Slave Timeout	ms
Master Ack Timeout(MAT)	ms, 0:DISABLE

☑ Save current setting to EEPROM
☑ Apply current setting

Set Remote VCOM3 connection

6. Select "**Add COM**" and type in the COM port of the PDS-700 #1 (Client) which you want to use.
7. Type in the COM port, IP and cmd port (default: 10000) of the **PDS-700 #2 (Server)**, and then check the "**Save to EEPROM**" and click "**submit**" button.



Address: http://10.0.8.5/

PDS-720 Remote Vcom3 connection Setup Page

#	Local COM port / Remote COM port
0	NONE

● Add COM 1 connect to COM 1 @ IP= 10.0.8.6 cmd port= 10000 (default:10000)
○ Delete #
○ Delete ALL

☑ Save to EEPROM
□ Reload from EEPROM

submit

8. Reboot your PDS-700 #1(Client) and then setting is complete.
9. Enter the configuration page of PDS-700#2 (Server) web server.
10. Click the "**COM Port Setting**" tag to enter the settings page of PDS-700#2 (Server), and then set the Baud Rate "**115200**" and data format **"8, None, 1**".
11. Click "**Set Remote VCOM3 connection**" at the PDS-700#2 (Server) COM PORT setup page.
12. Confirm that the Local COM port/Remote COM port field of the PDS-700#2 (Server) is "**None**".

Note!!
The baud rate and data format settings of the client and server (PDS-700 #1 and #2) are depending on COM ports of the connected device. The serial port settings can be different between the PDS-700 #1 and #2.

6. Modbus Testing and Protocol

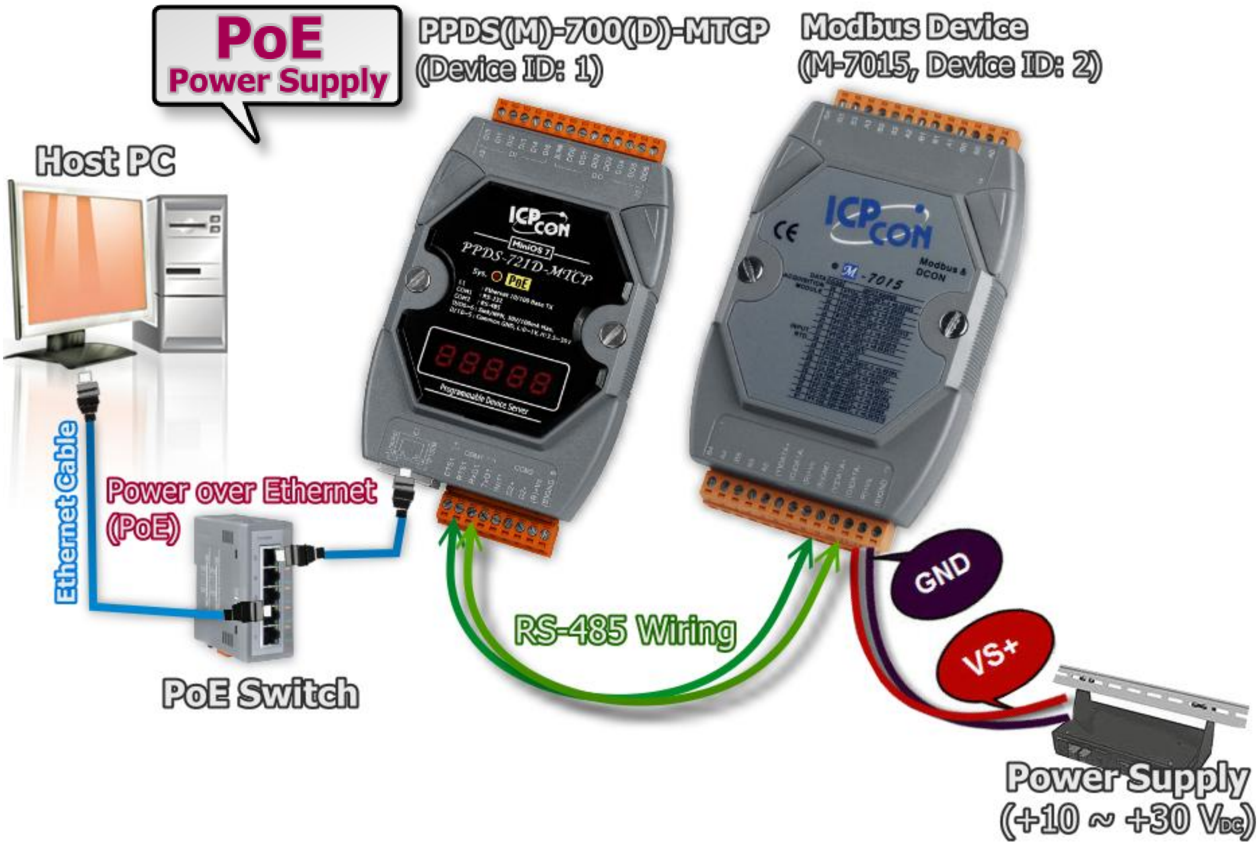
This chapter is only suitable for the PPDS (M) -700 (D)-MTCP, PDS-5105D-MTCP and PPDS-700-IP67.

Step1: Connecting the Modbus device to PDS series

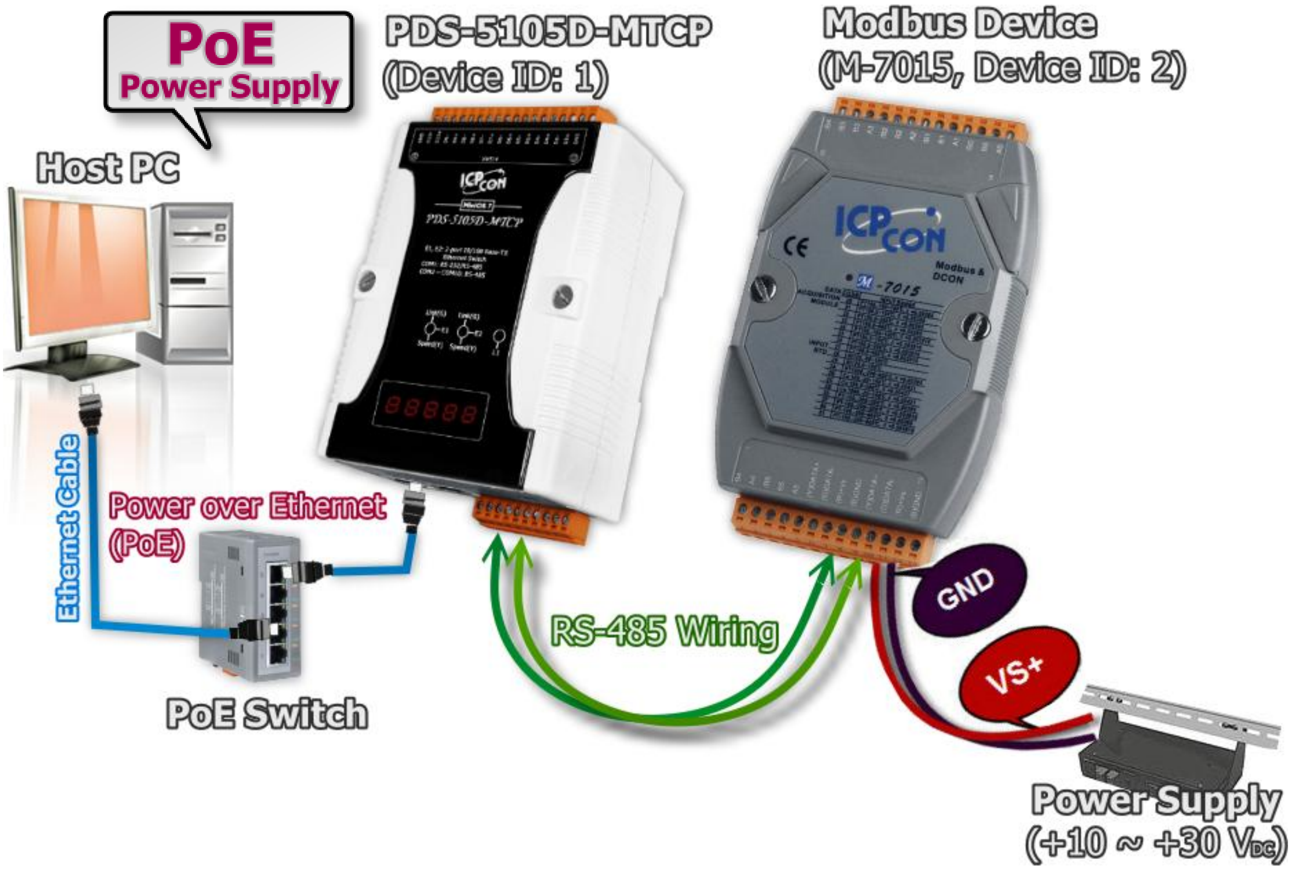
1. Keep up network connection status for your PPDS(M)-700(D)-MTCP, PDS-5105D-MTCP and PPDS-700-IP67. Refer to Sec. 4 Setting up the PDS for more details.
2. Connect the Modbus device (Ex: M-7015, optional) to PPDS(M)-700(D)-MTCP, PDS-5105D-MTCP and PPDS-700-IP67 on COM2 (RS-485 bus).
   (For detailed RS-232/422/485 wiring information, refer to Sec. 3.6 Wiring Notes.)
3. Supply power to the Modbus device. (M-7015, Device ID: 2, +10~+30 VDC Power used)

For wiring diagram as follows:

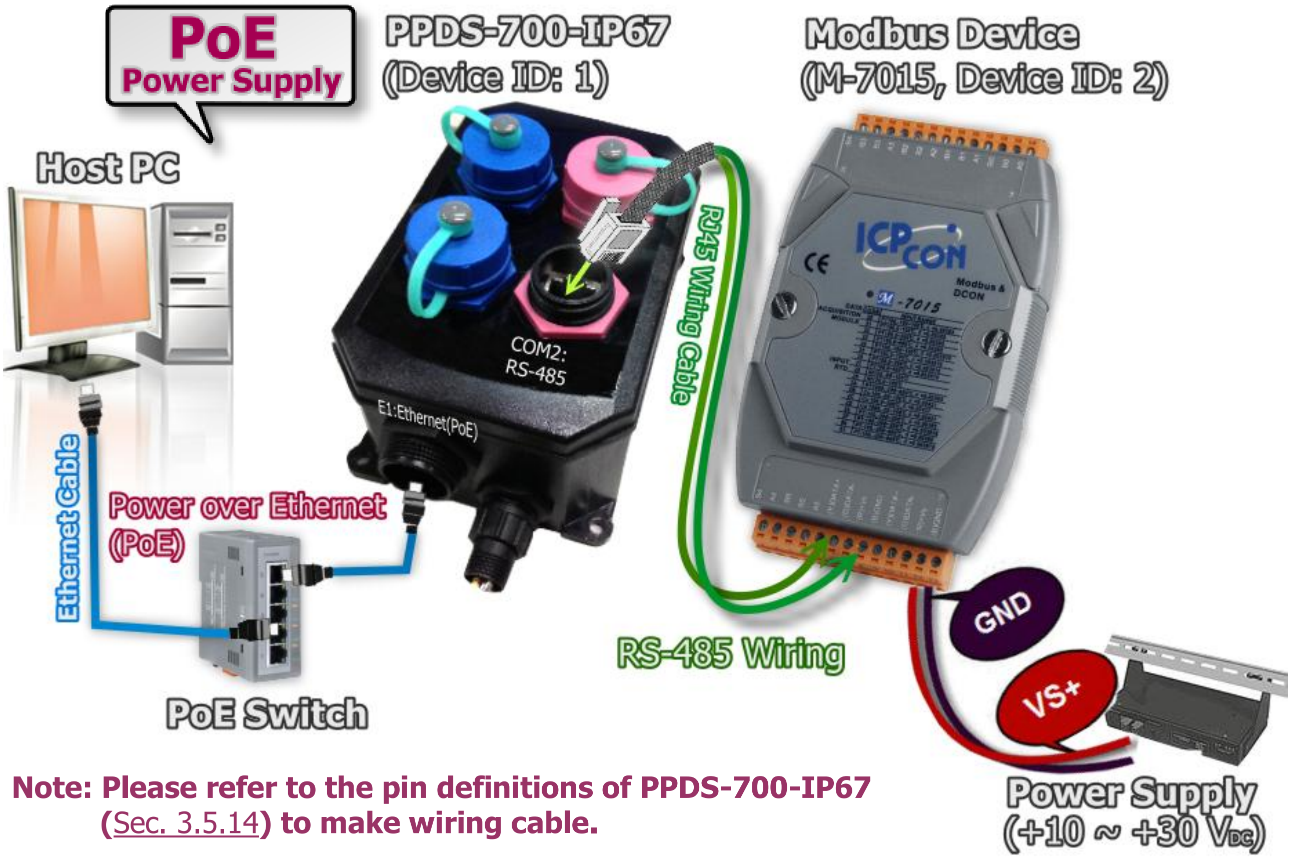
■ **PPDS(M)-700(D)-MTCP Series**



PDS-5105D-MTCP



PPDS-700-IP67 Series



Note: Please refer to the pin definitions of PPDS-700-IP67 (Sec. 3.5.14) to make wiring cable.

Step2: Install Modbus Utility on your PC

The software is located at:
CD: \NAPDOS\Modbus\modubs_utility
http://ftp.icpdas.com/pub/cd/8000cd/napdos/modbus/modbus_utility/

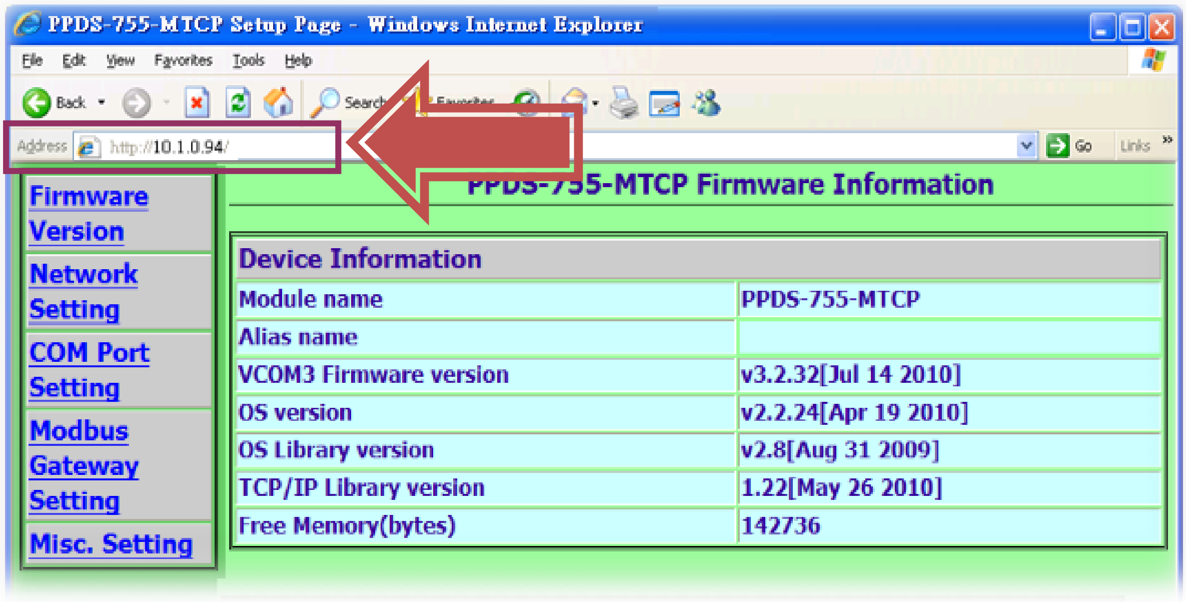
Modbus_Utility173_setup_20121015.exe

6.1 Modbus/TCP to Modbus/RTU Gateway

PPDS(M)-700(D)-MTCP, PDS-5105D-MTCP and PPDS-700-IP67 series can work as a Modbus/TCP to Modbus/RTU gateway that support most SCADA/HMI communications based on the Modbus/TCP protocol.

Step1: Configuring COM Ports for Modbus Gateway

1. Enter the IP address of the PPDS series module in the address field and press "**Enter**" to connect to the PPDS series module.

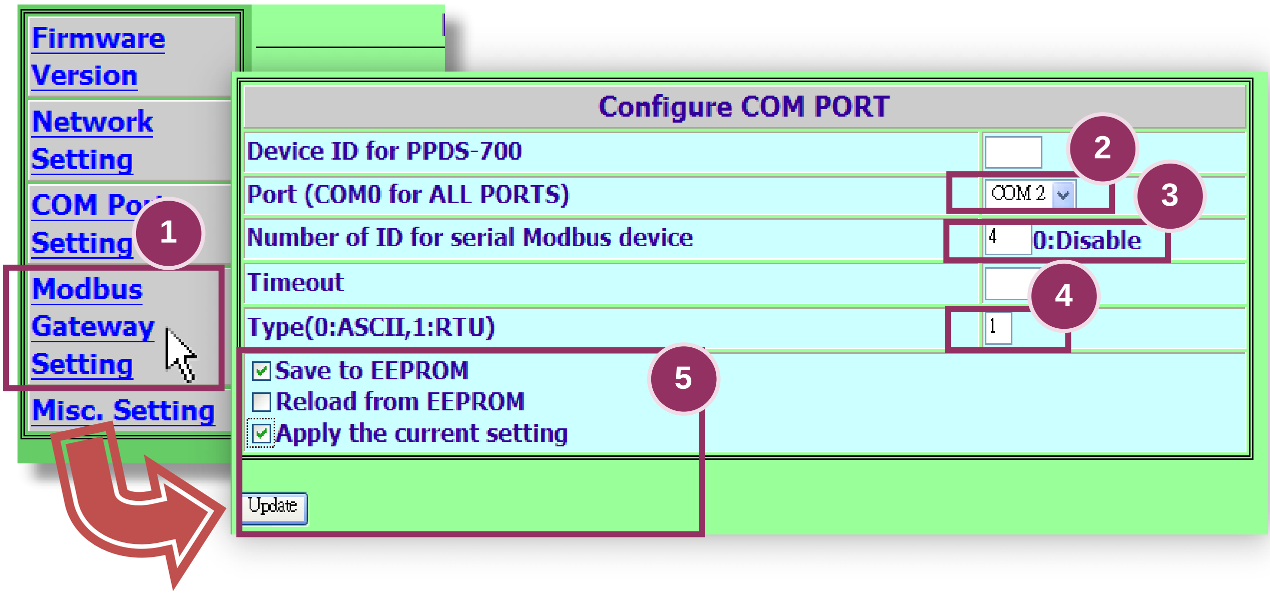


PPDS-755-MTCP Firmware Information

Device Information	
Module name	PPDS-755-MTCP
Alias name	
VCOM3 Firmware version	v3.2.32[Jul 14 2010]
OS version	v2.2.24[Apr 19 2010]
OS Library version	v2.8[Aug 31 2009]
TCP/IP Library version	1.22[May 26 2010]
Free Memory(bytes)	142736

2. Click the "**Modbus Gateway Setting**" in the web page to configure the COM ports, as follows:

   1: Select **appropriate COM Port** from the "**Port (COM0 for ALL PORTS)**" drop down options, for example "**COM2**".
   2: Assign a **Numbers of Modbus device** in the "**Number of ID for serial Modbus device**" field, for example "**4**".
   3: Select **Modbus type** from the "**Type (0: ASCII, 1: RTU)**" field, for example "**1(Modbus RTU)**".
   4: Checked the "**Save to EEPROM**" and "**Apply the current setting**" and then click the "**Update**" button to update it to PPDS series module.

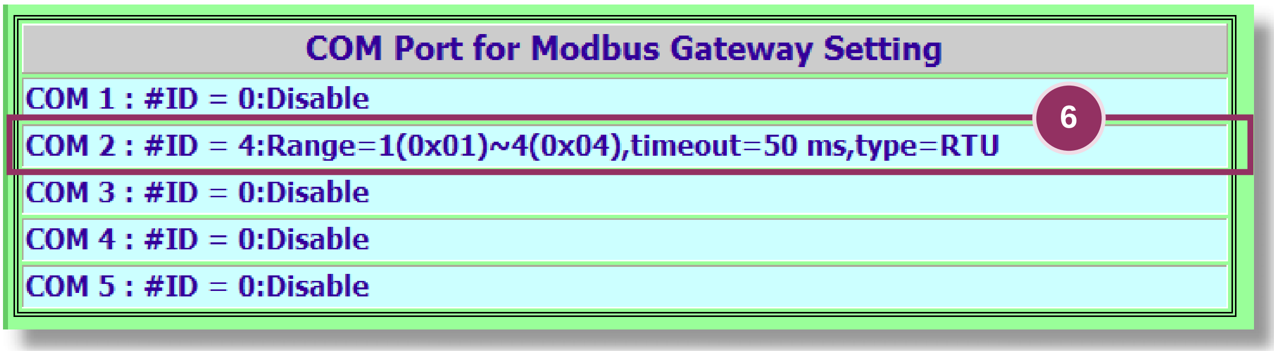


Configure COM PORT

Device ID for PPDS-700	
Port (COM0 for ALL PORTS)	COM 2
Number of ID for serial Modbus device	4 0:Disable
Timeout	
Type(0:ASCII,1:RTU)	1

☑ Save to EEPROM
□ Reload from EEPROM
☑ Apply the current setting

3. Check the COM Port for Modbus gateway setting.

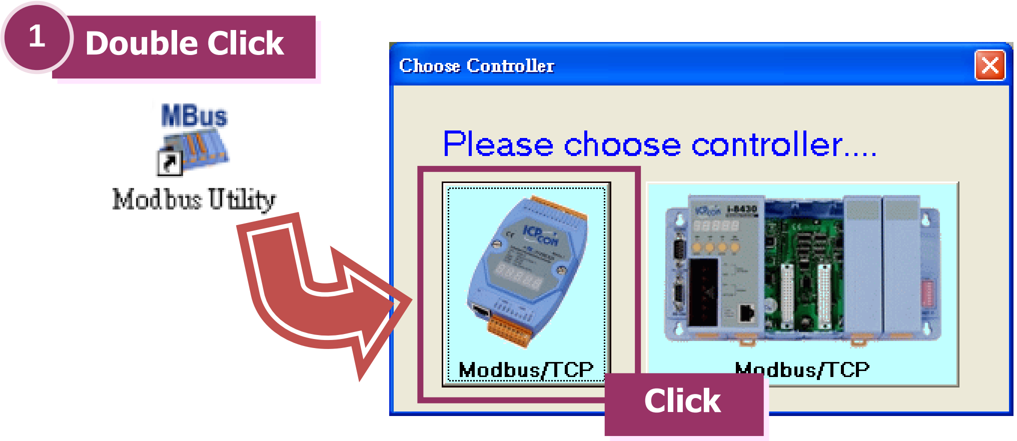


COM Port for Modbus Gateway Setting

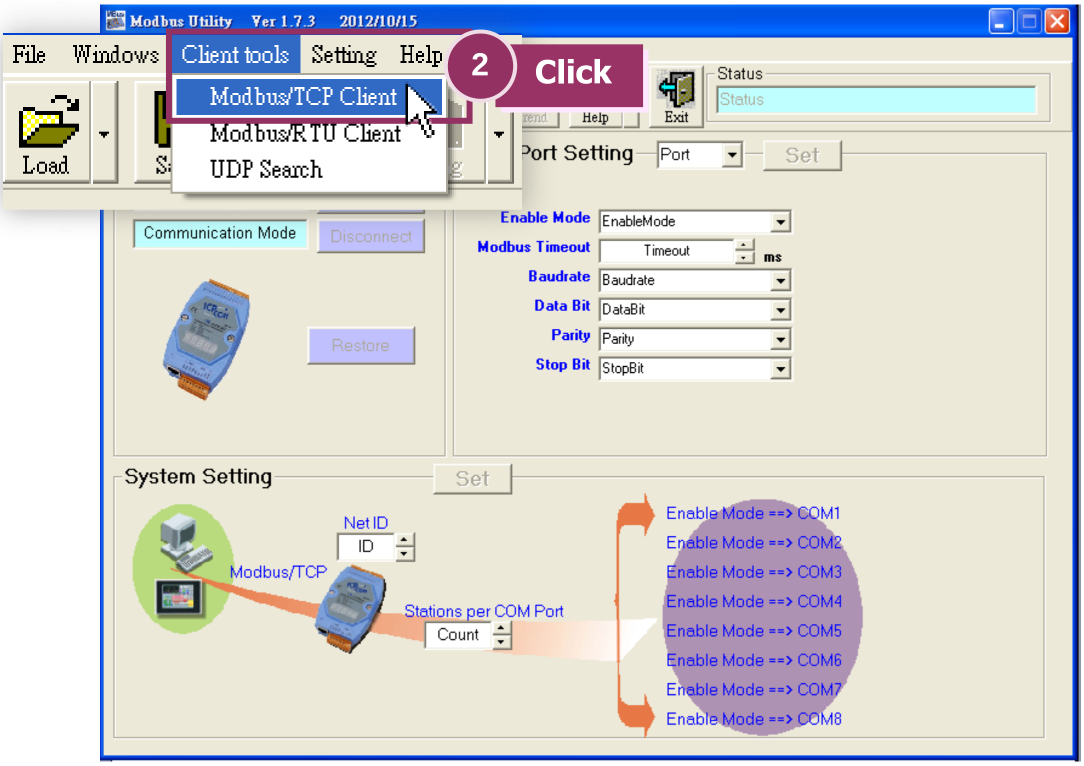
COM 1 : #ID = 0:Disable
COM 2 : #ID = 4:Range=1(0x01)~4(0x04),timeout=50 ms,type=RTU
COM 3 : #ID = 0:Disable
COM 4 : #ID = 0:Disable
COM 5 : #ID = 0:Disable

Step2: Test Modbus/TCP to Modbus/RTU Gateway

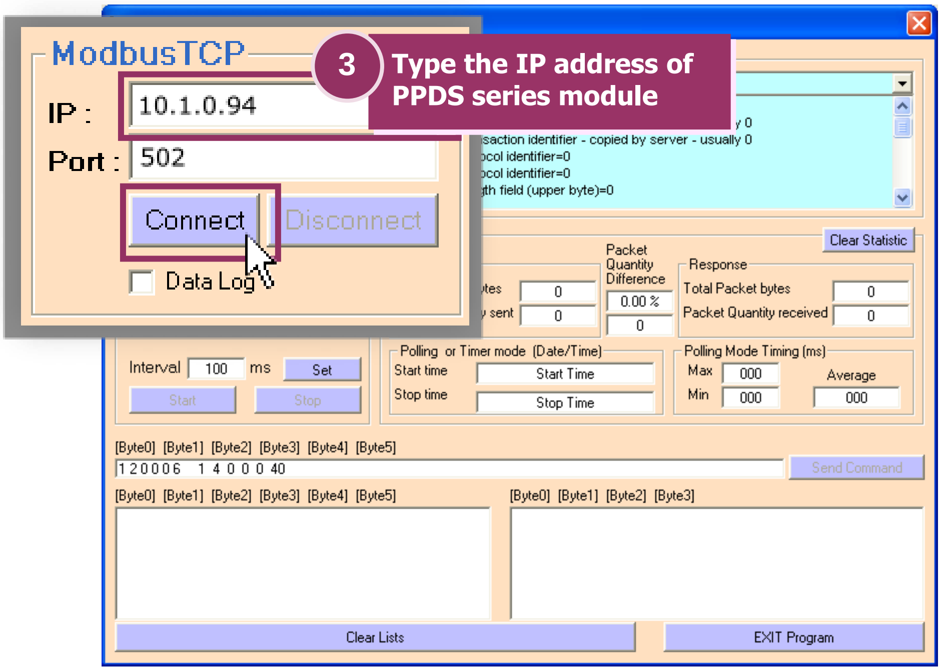
1. Run the Modbus Utility program and then click the **"Modbus/TCP"** button.



2. Click on "**Client tools**" → "**Modbus/TCP Client**".



3. Enter the IP address of PPDS series module and then click "**Connect**" button to connect the PPDS series module.



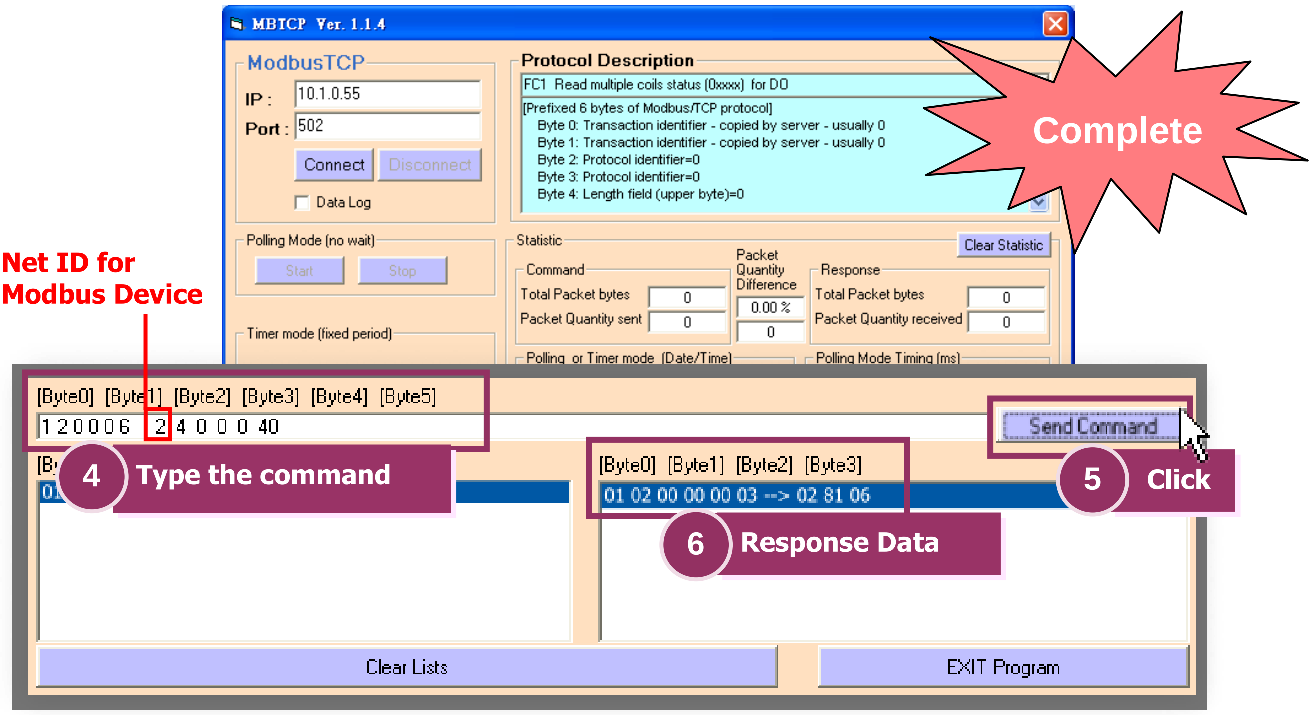
IP : 10.1.0.94

Port : 502

[Byte0] [Byte1] [Byte2] [Byte3] [Byte4] [Byte5]

```
1 2 0 0 0 6   1 4 0 0 0 40
```

4. Refer to "**Protocol Description**" and type command in the command field then click the "**Send Command**" button. If the response data is correct, it means the test is success.



MBTCP Ver. 1.1.4

IP : 10.1.0.55

Port : 502

Protocol Description

FC1 Read multiple coils status (0xxxx) for DO

[Prefixed 6 bytes of Modbus/TCP protocol]
Byte 0: Transaction identifier - copied by server - usually 0
Byte 1: Transaction identifier - copied by server - usually 0
Byte 2: Protocol identifier=0
Byte 3: Protocol identifier=0
Byte 4: Length field (upper byte)=0

[Byte0] [Byte1] [Byte2] [Byte3] [Byte4] [Byte5]

```
1 2 0 0 0 6   2 4 0 0 0 40
```

[Byte0] [Byte1] [Byte2] [Byte3]

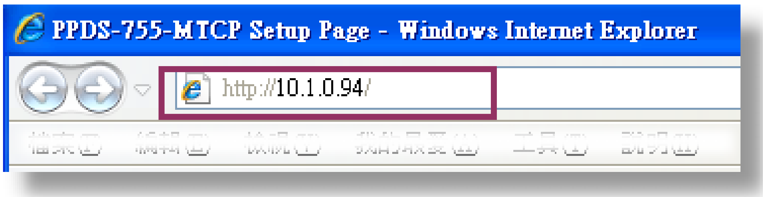
```
01 02 00 00 00 03 --> 02 81 06
```

6.2 Testing Modbus Device through Virtual COM Ports

If want to use Modbus/RTU through Virtual COM Ports, you can refer to below steps.

Step1: Configuring COM Ports for Virtual COM

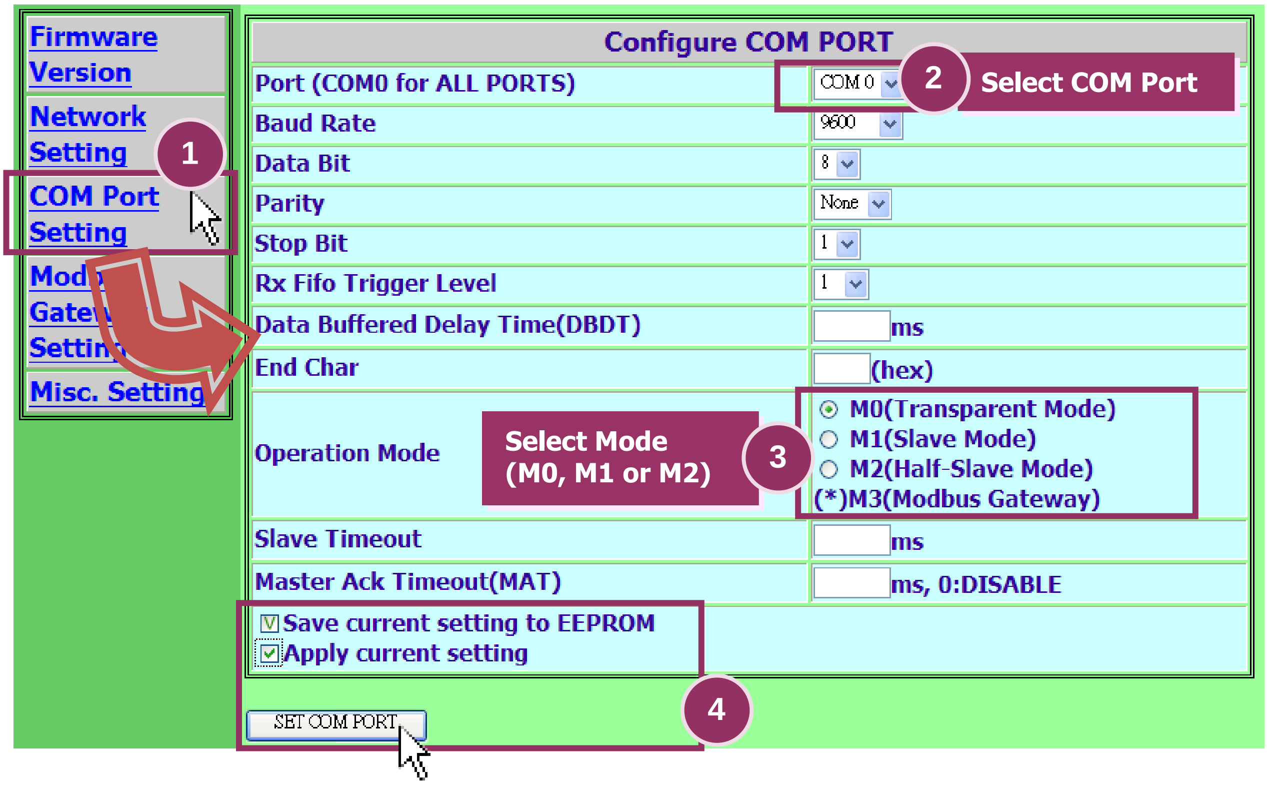
1. Enter the IP address of the PPDS series module in the address field and press "**Enter**" to connect to the PPDS series module.



PPDS-755-MTCP Setup Page - Windows Internet Explorer

http://10.1.0.94/

2. Make sure your COM Ports of PPDS series module which aren't in the M3 mode. Select "**COM Port Setting**" in the web configuration and set the "**Operation Mode**". Checked the "**Save current settings to EEPROM**", "**Apply current setting**" and then click the "**SET COM PORT**" button to complete the setting.



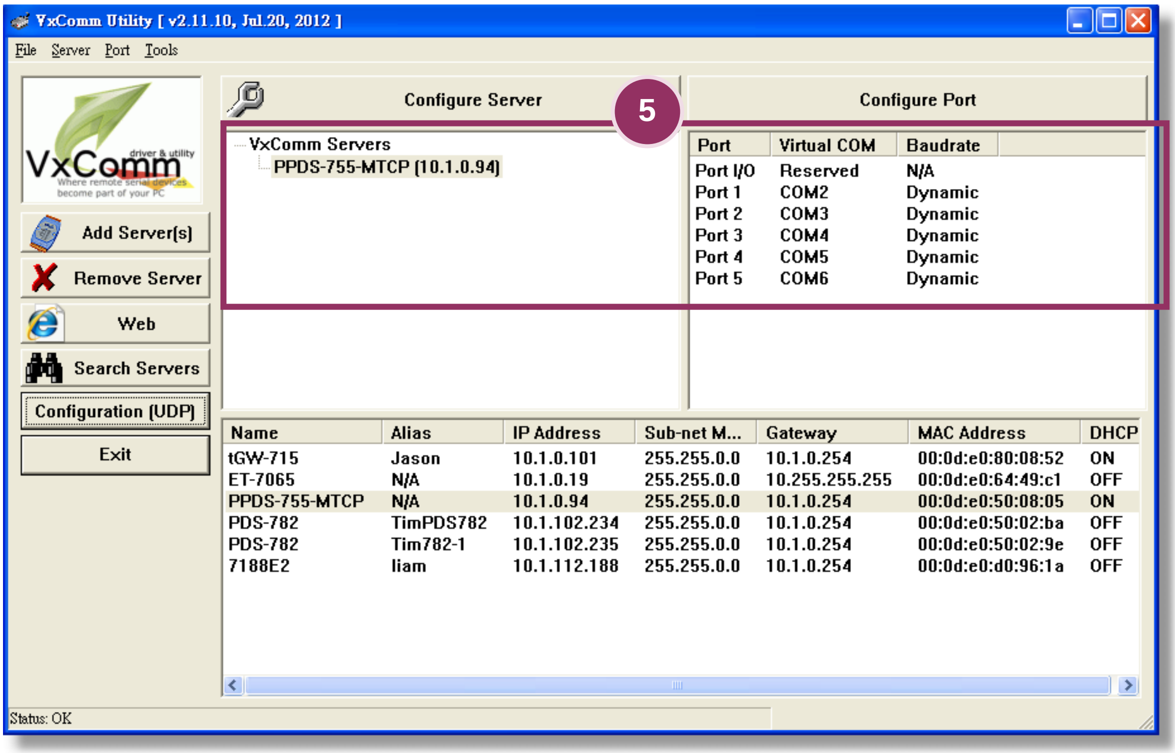
Configure COM PORT

Port (COM0 for ALL PORTS)	COM 0
Baud Rate	9600
Data Bit	8
Parity	None
Stop Bit	1
Rx Fifo Trigger Level	1
Data Buffered Delay Time(DBDT)	ms
End Char	(hex)
Operation Mode	● M0(Transparent Mode) ○ M1(Slave Mode) ○ M2(Half-Slave Mode) (*)M3(Modbus Gateway)
Slave Timeout	ms
Master Ack Timeout(MAT)	ms, 0:DISABLE

☑ Save current setting to EEPROM
☑ Apply current setting

SET COM PORT

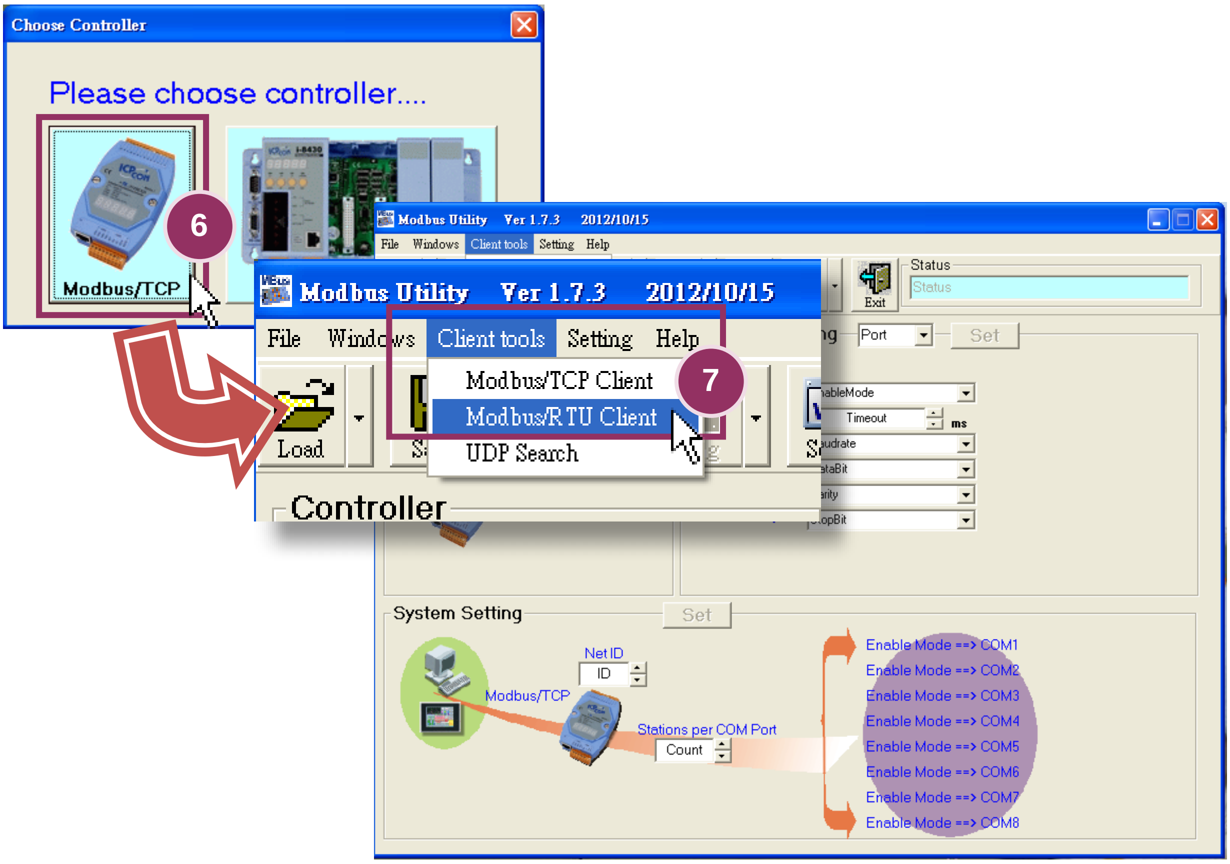
3. Add Server(s) in the VxComm Utility please refer to Section 4 Setting up the PDS Module.



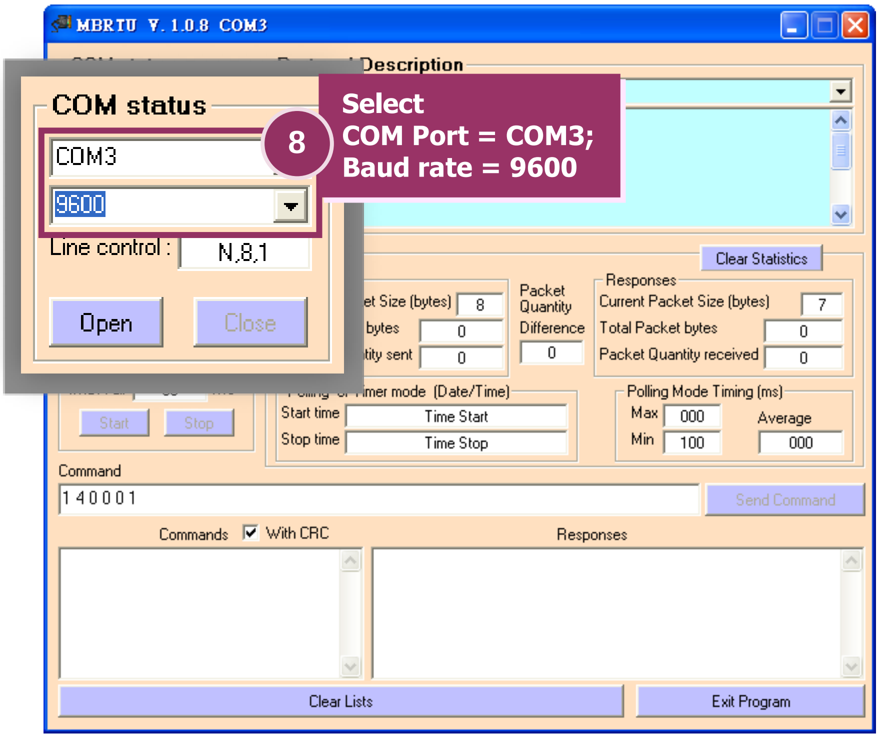
Port	Virtual COM	Baudrate
Port I/O	Reserved	N/A
Port 1	COM2	Dynamic
Port 2	COM3	Dynamic
Port 3	COM4	Dynamic
Port 4	COM5	Dynamic
Port 5	COM6	Dynamic

Name	Alias	IP Address	Sub-net M...	Gateway	MAC Address	DHCP
tGW-715	Jason	10.1.0.101	255.255.0.0	10.1.0.254	00:0d:e0:80:08:52	ON
ET-7065	N/A	10.1.0.19	255.255.0.0	10.255.255.255	00:0d:e0:64:49:c1	OFF
PPDS-755-MTCP	N/A	10.1.0.94	255.255.0.0	10.1.0.254	00:0d:e0:50:08:05	ON
PDS-782	TimPDS782	10.1.102.234	255.255.0.0	10.1.0.254	00:0d:e0:50:02:ba	OFF
PDS-782	Tim782-1	10.1.102.235	255.255.0.0	10.1.0.254	00:0d:e0:50:02:9e	OFF
7188E2	liam	10.1.112.188	255.255.0.0	10.1.0.254	00:0d:e0:d0:96:1a	OFF

4. Run Modbus Utility program and click the **"Modbus/TCP"** button and then click on **"Client tools"** → **"Modbus/RTU Client"**.



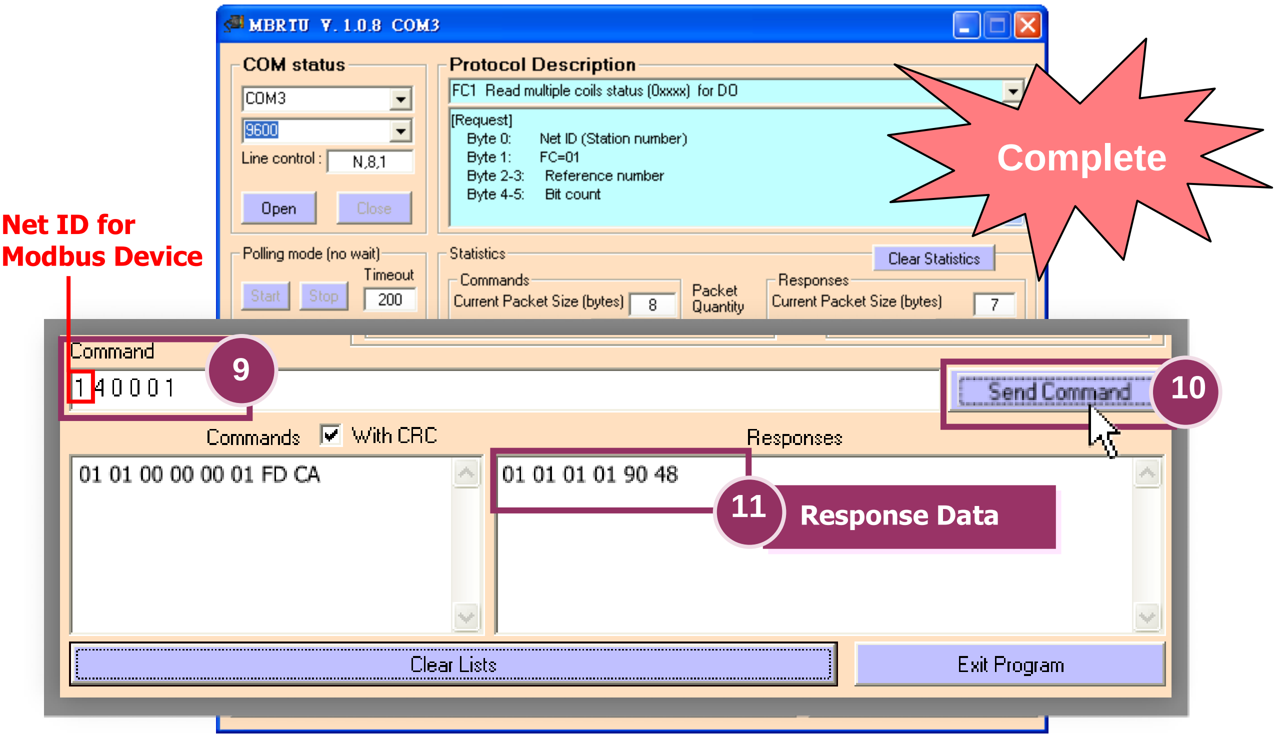
5. Select your virtual COM port and baud rate (Default: 9600) on PPDS series module, and then click the "**Open**" button.



COM3
9600

Select
COM Port = COM3;
Baud rate = 9600

6. Refer to "**Protocol Description**" and type the command in the command field then click the "**Send Command**" button. If the response data is correct, it means the test is success.



FC1 Read multiple coils status (0xxxx) for DO

[Request]
Byte 0: Net ID (Station number)
Byte 1: FC=01
Byte 2-3: Reference number
Byte 4-5: Bit count

Command
1 4 0 0 0 1

Commands ☑ With CRC
01 01 00 00 00 01 FD CA

Responses
01 01 01 01 90 48

7. Virtual I/O

PDS series modules provide digital I/O lines, including PDS(M)-721(D), PPDS(M)-721(D)-MTCP, PDS(M)-732(D), PPDS(M)-732(D)-MTCP, PDS(M)-734(D), PPDS(M)-734(D)-MTCP, PDS(M)-743(D), PPDS(M)-743(D)-MTCP, PDS(M)-762(D) and PPDS(M)-762(D)-MTCP. The DI is 0 ~ 30 V_{DC} wide range Digital Input, while the DO is 30 V/100 mA (max.), current sink, open collector digital output. These digital I/O lines can be used to control relays, actuators, switches, etc.

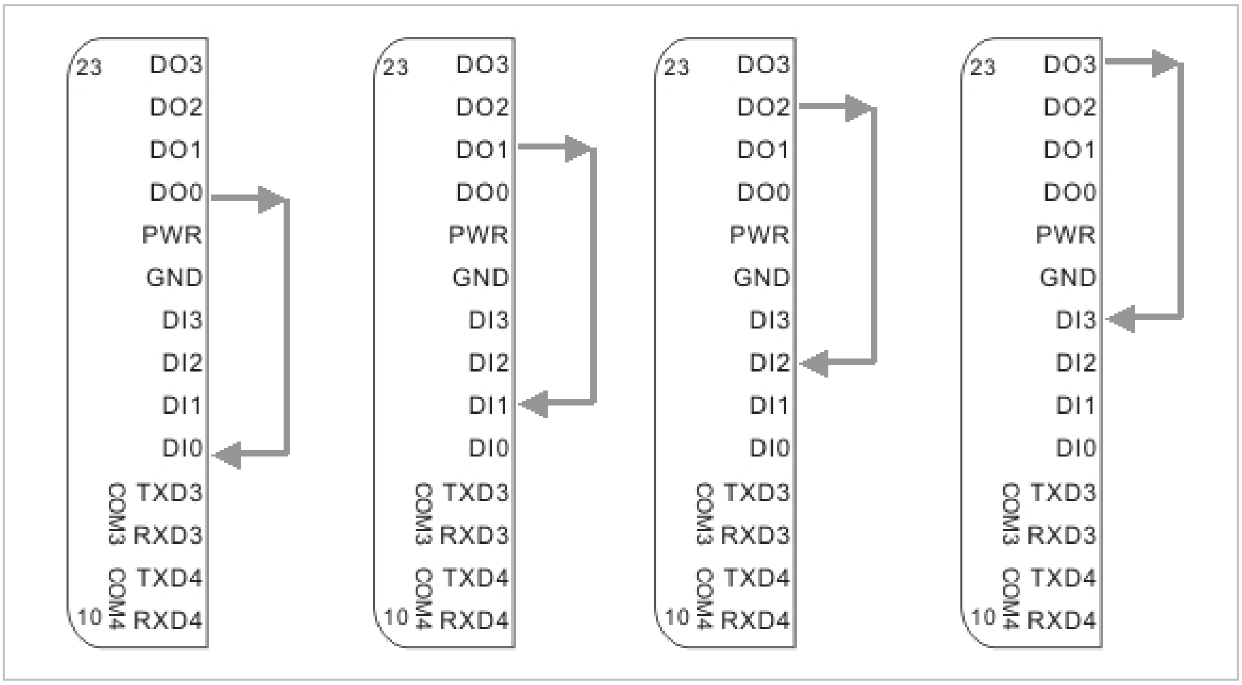
7.1 Testing the Virtual I/O

1. Connect the PDS series module to the Ethernet, finalize the configuration setup procedure and complete the Virtual COM test, as described in Chapter 4.

2. Power-on the PDS series module.

3. Connect the DO (n) to the DI (n).

For example, the PDS-734 with a 4-port DI/DO.

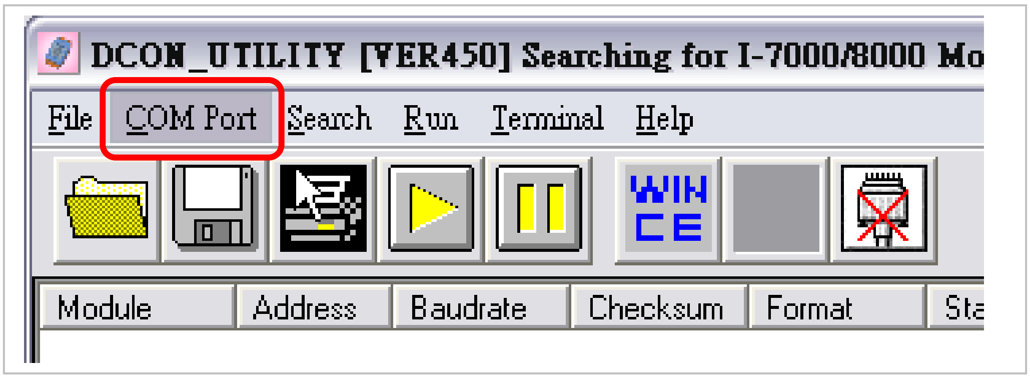


Connect DO 0 to DI 0, DO 1 to DI 1, DO 2 to DI 2 and DO 3 to DI 3.

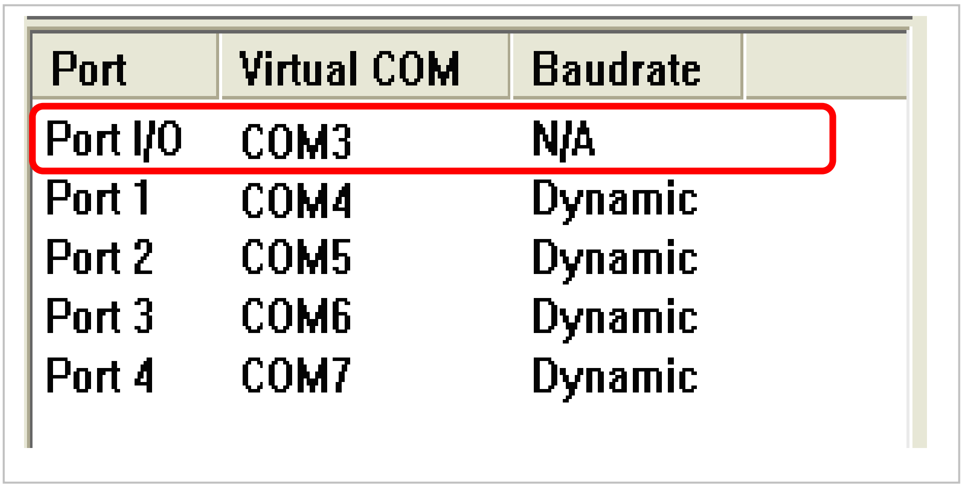
4. Install the DCON Utility v4.5.0 (or later).

The DCON Utility is located at:
CD:\Napdos\driver\dcon_utility\
http://ftp.icpdas.com/pub/cd/8000cd/napdos/driver/dcon_utility/setup/

5. Run the DCON Utility, and click the "**COM Port**" option on the toolbar

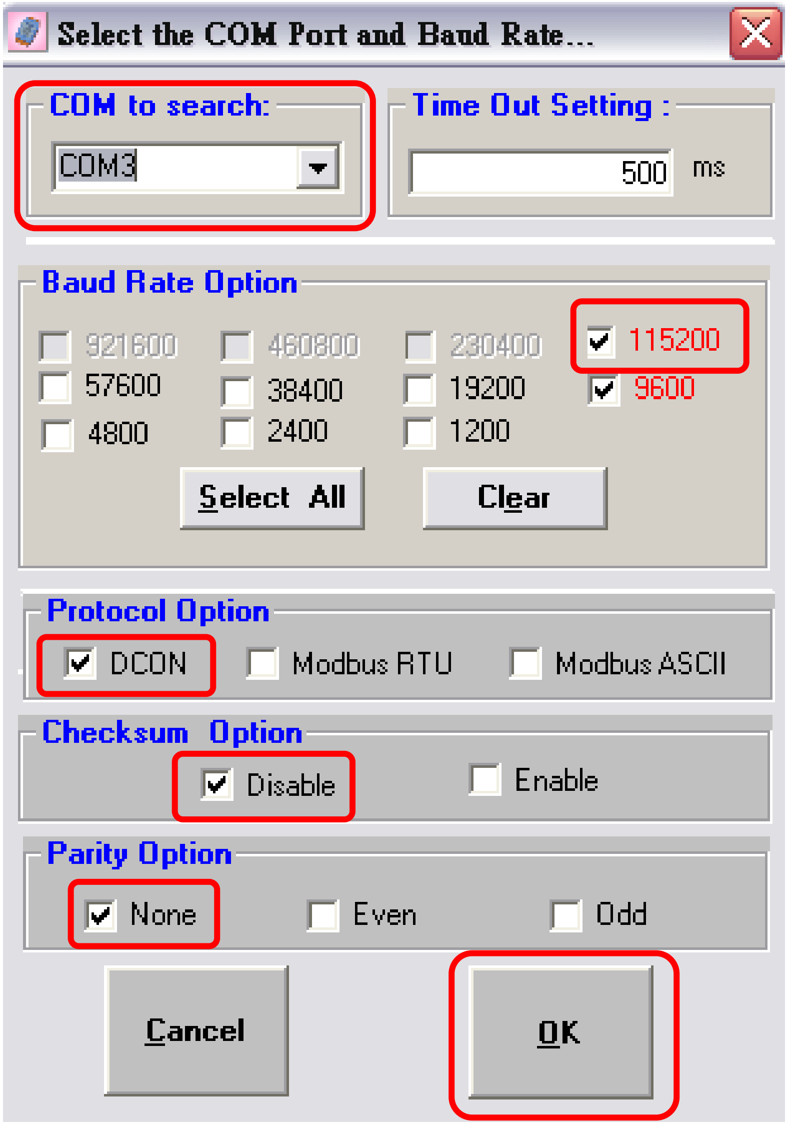


6. Check the Virtual COM Port number shown in the Port I/O field in the VxComm Utility.(Refer to Chapter 4 for more details)



Port	Virtual COM	Baudrate
Port I/O	COM3	N/A
Port 1	COM4	Dynamic
Port 2	COM5	Dynamic
Port 3	COM6	Dynamic
Port 4	COM7	Dynamic

7. Select the Virtual COM Port number. Check 115200 as the Baud Rate, DCON as the protocol, checksum disabled, parity as none, and then click the "**OK**" button.



Select the COM Port and Baud Rate…

COM to search: COM3

Time Out Setting : 500 ms

Baud Rate Option

□ 921600 □ 460800 □ 230400 ☑ 115200
□ 57600 □ 38400 □ 19200 ☑ 9600
□ 4800 □ 2400 □ 1200

Protocol Option

☑ DCON □ Modbus RTU □ Modbus ASCII

Checksum Option

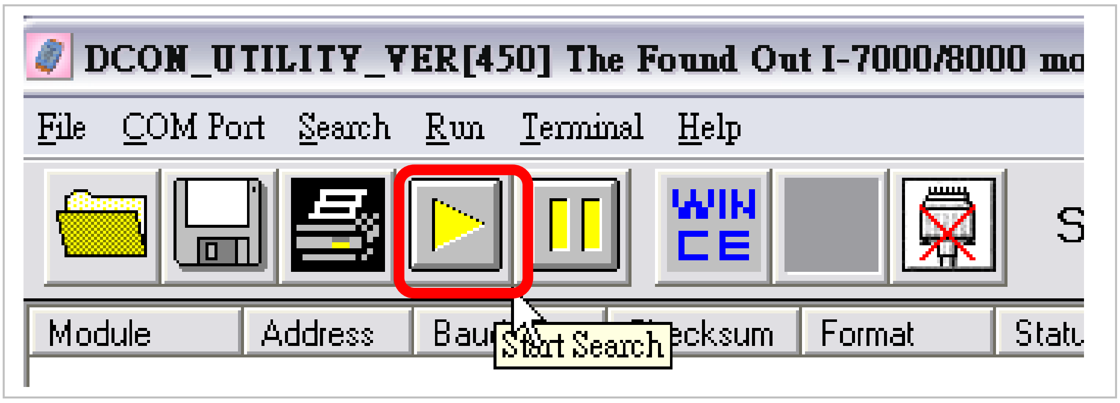
☑ Disable □ Enable

Parity Option

☑ None □ Even □ Odd

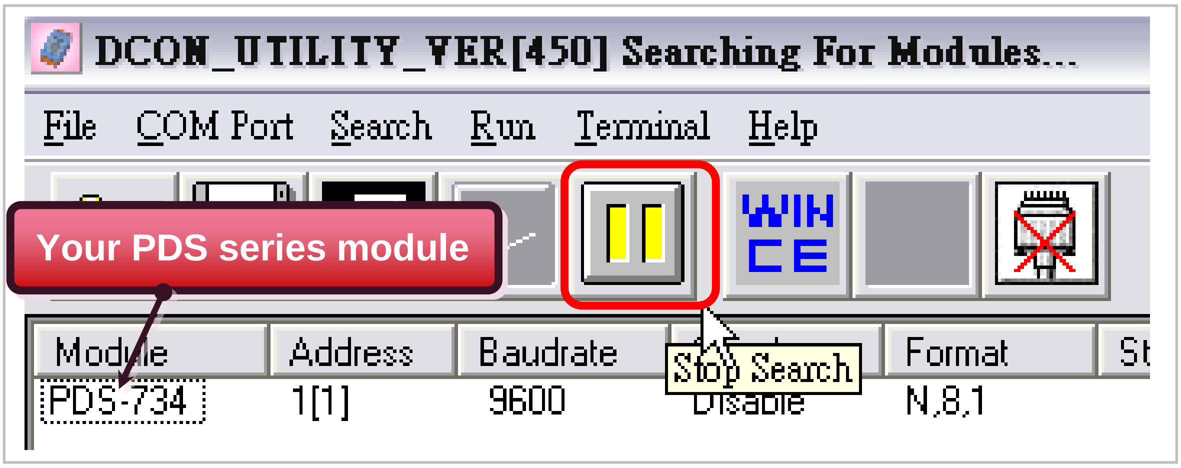
*If your PDS is not equipped with digital I/O lines, the DCON Utility will return an "**Open COM error!**" message.*

8. Click the ▶ button to start searching for the PDS series module


DCON_UTILITY_VER[450] The Found Out I-7000/8000 mo

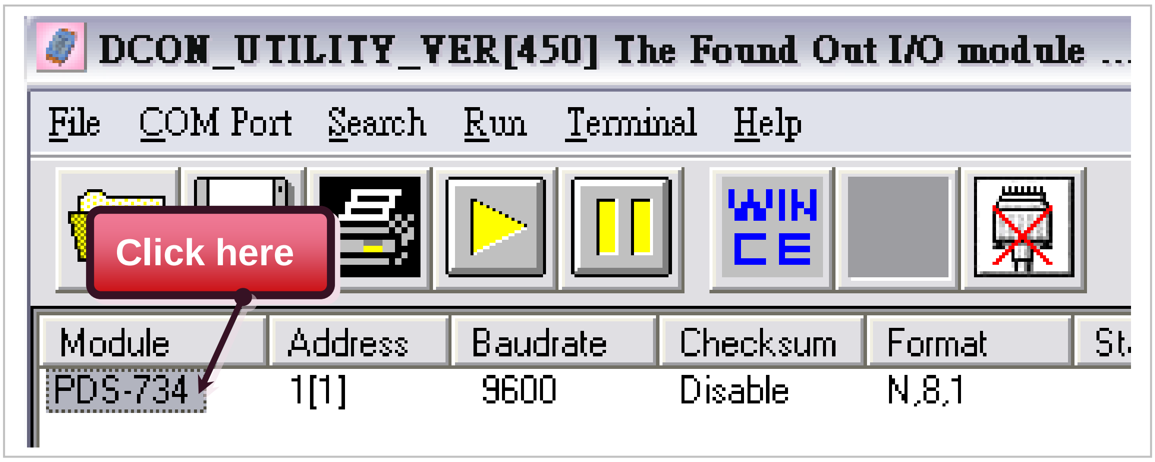
Module	Address	Bau	ecksum	Format	Statu

9. When the PDS series module is found and is displayed in the DCON Utility, click the ❚❚ button to stop the search


DCON_UTILITY_VER[450] Searching For Modules...

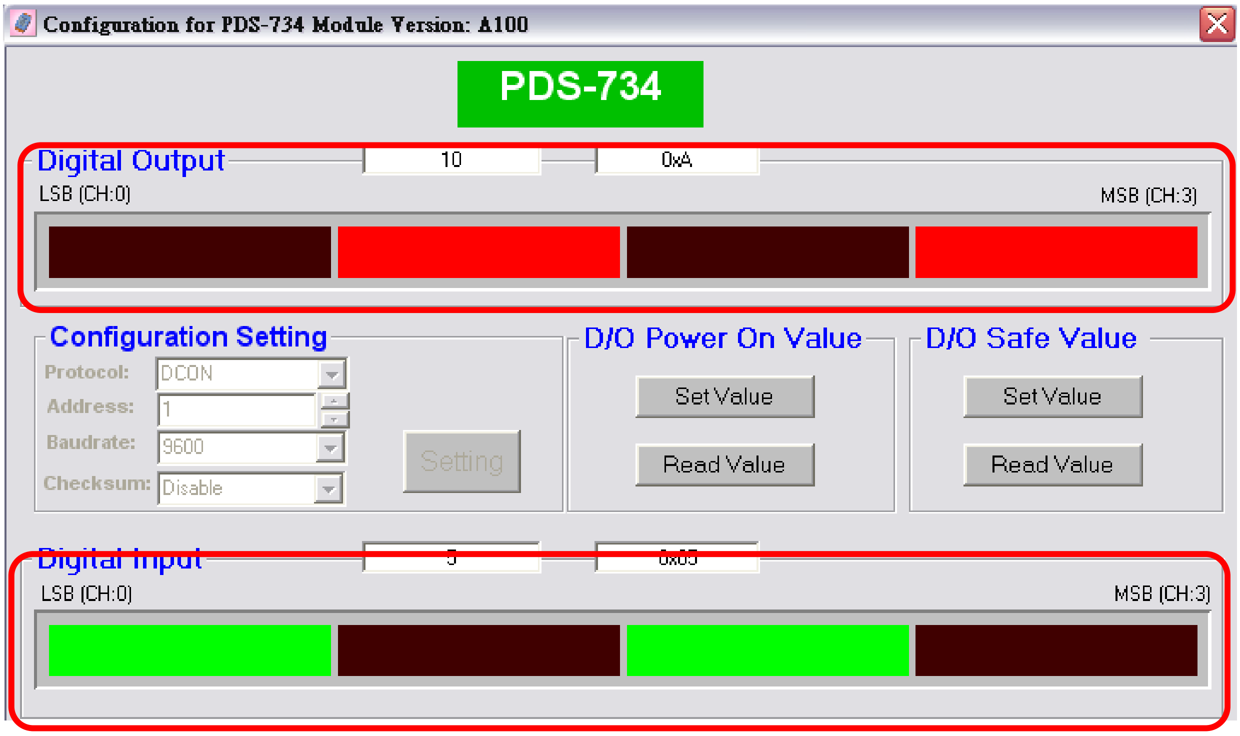
Module	Address	Baudrate		Format	St
PDS-734	1[1]	9600	Disable	N,8,1	

10. Click on the name of your PDS series module.


DCON_UTILITY_VER[450] The Found Out I/O module ...

Module	Address	Baudrate	Checksum	Format	St
PDS-734	1[1]	9600	Disable	N,8,1	

11. Click the “**Digital Output**” icon to change the high/low status of the DO.



Configuration for PDS-734 Module Version: A100

PDS-734

Digital Output: 10 0xA

Configuration Setting

Protocol: DCON

Address: 1

Baudrate: 9600

Checksum: Disable

Digital Input: 5 0x05

Since all DI lines are connected to DO lines, the DI read value will be 0 when the DO sends a high state, where as the DI read value will be 1.

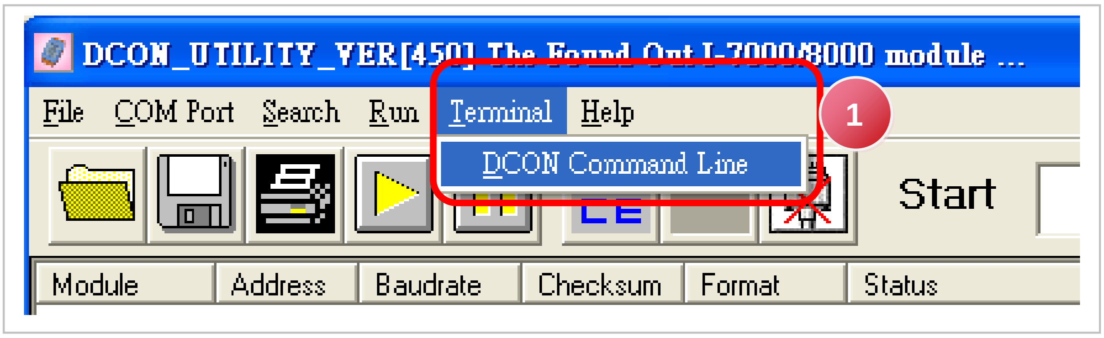
7.2 Virtual I/O Commands Test

The DCON protocol is a request/reply communication protocol; it defines a simple ASCII format protocol, such as $AAN, $AASi6 and #AAN, etc. used to access the PDS and I-7000/8000/87K series I/O modules.

The Virtual I/O command sets are part of the DCON protocol used to access the digital I/O lines of the PDS from the virtualized COM Port mapped to the I/O port. Only PDS series modules equipped with digital I/O lines will respond to DCON requests.

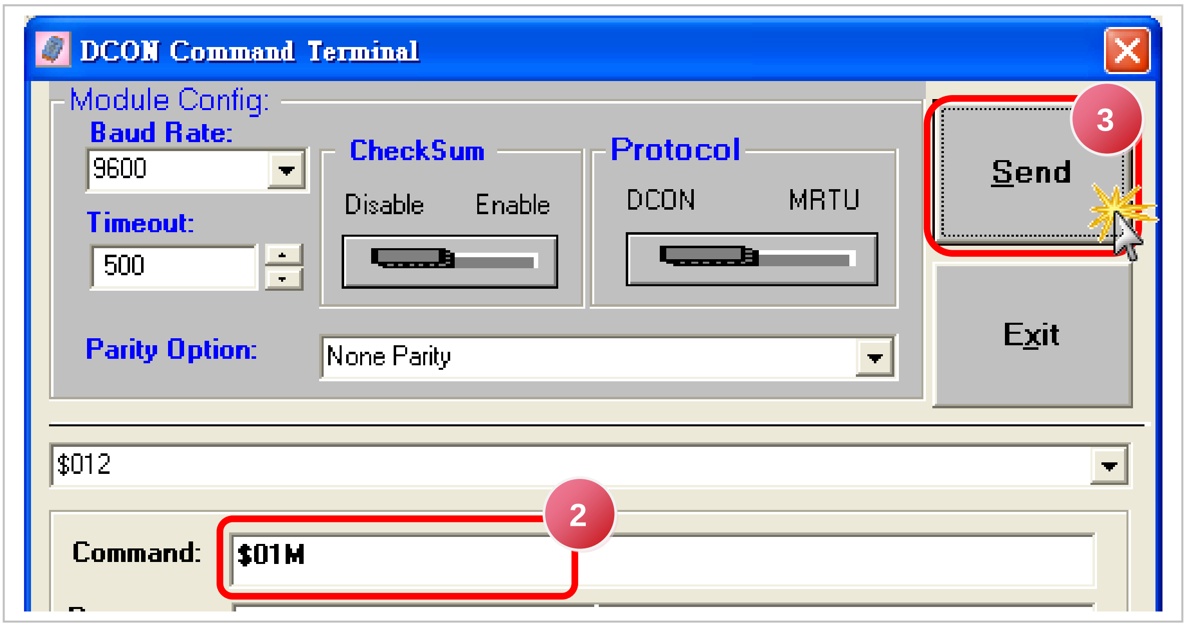
The DCON Utility can be used to test the Virtual I/O commands:
(The DCON command sets are introduced in Chapter 7)

1. Select "**Terminal**" >> "**DCON Command Line**" from the DCON Utility menu.



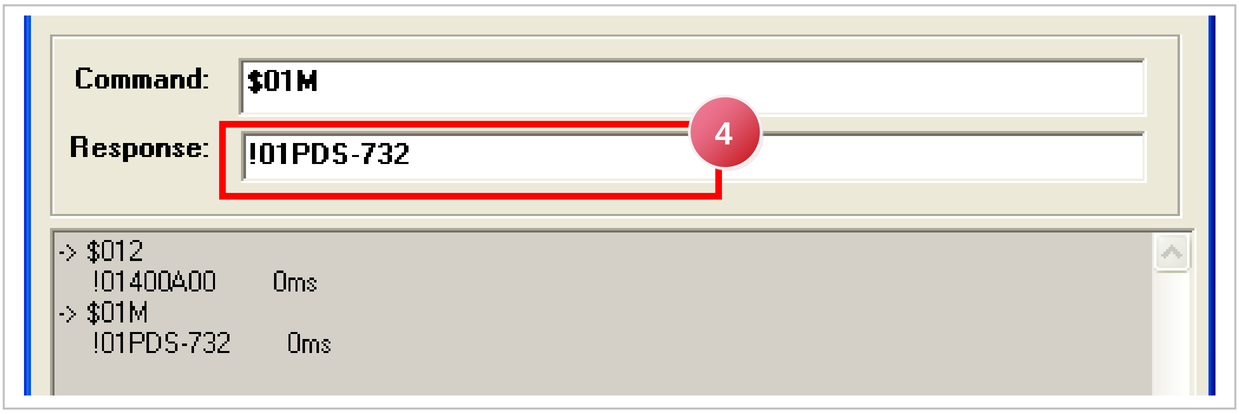
2. Type the Virtual I/O command in the command column and click the "**Send**" button to send the command.

For example, the command $01M is used to read the module name.



Baud Rate: 9600
Timeout: 500
Parity Option: None Parity
$012
Command: $01M

3. Receive a response from the PDS module that the command was sent successfully.



Command: $01M
Response: !01PDS-732

```
-> $012
   !01400A00      0ms
-> $01M
   !01PDS-732     0ms
```

7.3 Programming on a PC client

The General DCON Application Programming Interface kit is a set of DLL (lib) functions designed to run on Windows 98/2000/XP that allow access to remote I/O modules such as the PDS-700, I-7000, I-8000 and I-87k series.

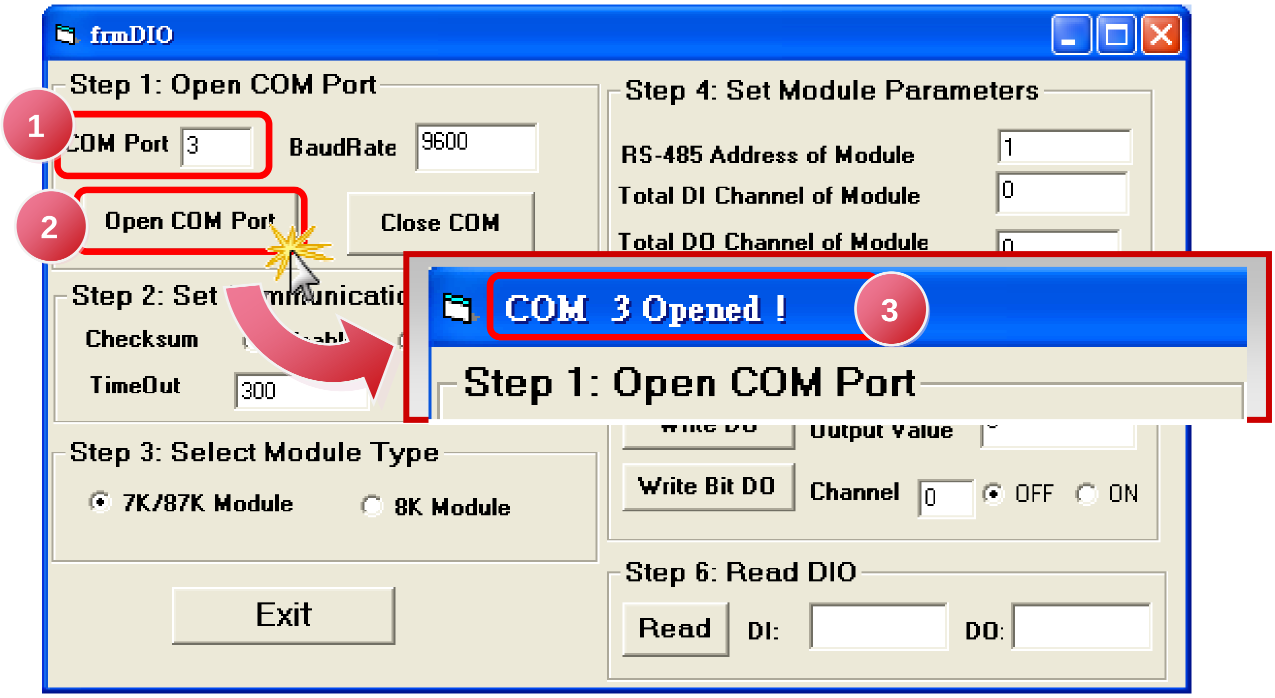
The General DCON API kit is located at:

- CD:\ napdos\ driver\ dcon_dll_new\
- ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/driver/dcon_dll_new/

The General DCON API kit provides VC and VB drivers, VB demos and a document called "**dcon_fun_user_manual.pdf**". Only the DIO demo that can be found in the dcon_dll_new\demo\vb6 folder supports PDS series modules. The following steps can be used to test the general DCON API kit with the DIO demo programs.

To run the DIO demo, VB6 must first be installed on the PC.

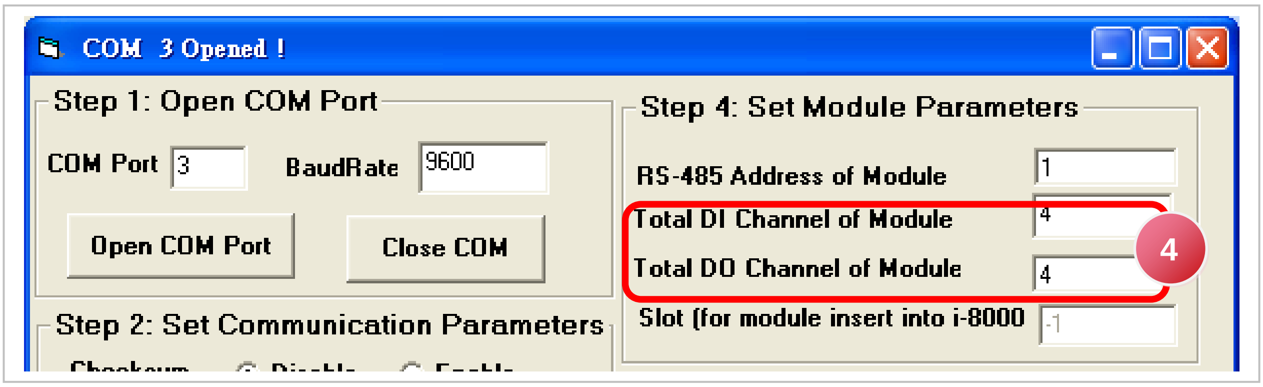
1. Double click "**prjdio.vbp**" to open the DIO project.
2. Run the demo.
3. Set the Virtual COM Port number of the PDS and click the "**Open COM Port**" button. The response "**COM n Opened!**" will be shown on the title bar.



COM 3 Opened !

COM Port: 3
BaudRate: 9600
TimeOut: 300
● 7K/87K Module ○ 8K Module
RS-485 Address of Module: 1
Total DI Channel of Module: 0
Total DO Channel of Module: 0
Channel: 0 ● OFF ○ ON

4. Set the total number of DI and DO channels on your PDS series module.
   For instance, the PDS-732 is equipped with 4 DI channels and 4 DO channels.


COM 3 Opened !

Step 1: Open COM Port

COM Port: 3

BaudRate: 9600

Step 2: Set Communication Parameters

Step 4: Set Module Parameters

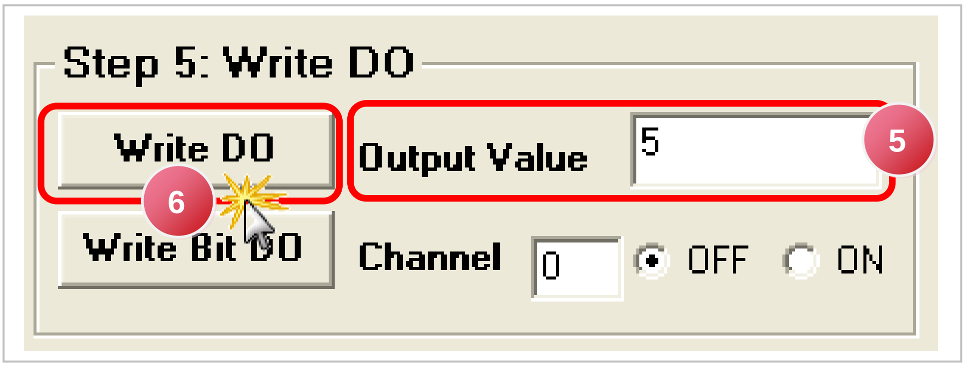
RS-485 Address of Module: 1

Total DI Channel of Module: 4

Total DO Channel of Module: 4

Slot (for module insert into i-8000: -1

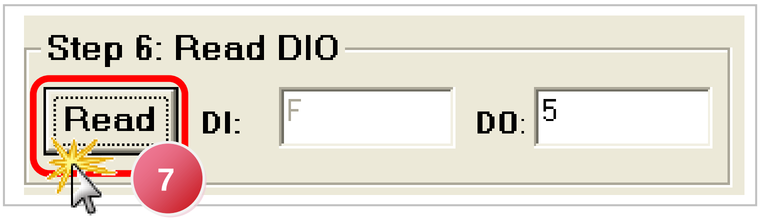
5. Set the Output value and then click the "**Write DO**" button to transmit the data.


Step 5: Write DO

Output Value: 5

Channel: 0 ● OFF ○ ON

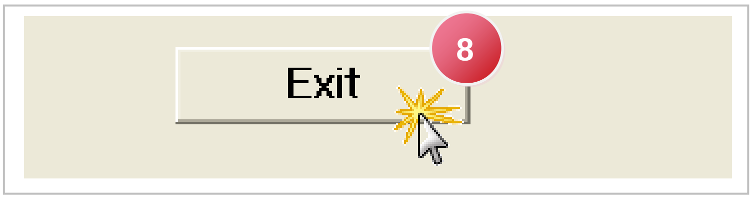
6. Click the "**Read**" button to retrieve the DI data and read the DO data.


Step 6: Read DIO

DI: F

DO: 5

7. Press the "**Exit**" button to exit the program.



The functions in the General DCON API kit can be used to access the I/O lines on the PDS series module:

Categorization		Dll and lib	Call condition
Sec. 7.3.1	**Starting function** O pen_Com()	Uart.dll Uart.lib	Called once when the program starts

Sec. 7.3.4
Sec. 7.3.5
Sec. 7.3.6
	I/O function
	DCON_Write_DO()
	DCON_Write_DO_Bit()
	DCON_Read_DIO()
		dcon_pc.dll
		dcon_pc.lib
			Calls the I/O functions for requirements

Sec. 7.3.3	**Communication** Send_Receive_Cmd()	Uart.dll Uart.lib	Calls the communication functions for requirements
Sec. 7.3.2	**Ending function** Close_Com()	Uart.dll Uart.lib	Called once before the program exits

```
// DO program demo on a PC client
void CManual1Dlg::OnOpen_Com()
{
      Open_Com(3,115200,8,0,1);
      //COM Port: 3, Baud Rate:115200, Data Bit:8, Parity Bit: 0, Stop Bit: 1
}

void CManual1Dlg::OnClose_Com()
{
     Close_Com(3);  }

void CManual1Dlg::OnDigital_Out()
{
   iRet=DCON_Write_DO(3,1,-1,4,iDO_value,0,100);
      //COM Port: 3, Address: 1, Slot: -1, total channel count:4, DO data,
      //Checksum: disabled, Timeout: 100 (ms)
}
```

7.3.1 Open_Com()

- **Description:**
  This function opens the specified COM Port.

- **Syntax:**
  Open_Com(unsigned char **cPort**, DWORD **dwBaudrate**, char **cData**, char **cParity**, char **cStop**);

- **Parameters:**

cPort	COM Port number (1 ~ 255)
dwBaudrate	Communication Baud Rate
cData	Data bit, (8 for PDS)
cParity	0 = No parity
cStop	0 = 1 Stop bit

- **Return:**
  0 → no error
  Others → error codes

7.3.2 Close_Com ()

- **Description:**
  This function closes the specified COM Port.

- **Syntax:**
  Close_Com(unsigned char **cPort**)

- **Parameters:**

cPort	COM Port number (1 ~ 255)

- **Return:**
  0 → no error
  Others → error codes

7.3.3 Send_Receive_Cmd ()

- **Description:**

This function sends a DCON command string and receives the response.

- **Syntax:**

Send_Receive_Cmd(unsigned char **cPort**, char **szCmd[ ]**, char **szResult[ ]**,
WORD **wTimeOut**, WORD **wChksum**, WORD ***wT**)

- **Parameters:**

cPort	COM Port number (1 ~ 255)
szCmd[ ]	the send string, 1024 bytes maximum, without a zero (0x0D) character
szResult[ ]	the result string recevied, 1024 bytes maximum, with one zero or 0x0D terminal character
wTimeOut	timeout for receiving the result string. Unit: ms
wChksum	**0** → add one 0x0D byte to the end of the szCmd **<>0** → add two check sum bytes and one 0x0D byte to the end of the szCmd
***wT**	return a reference number to identify the performance

- **Return:**

0 → no error

Others → error codes

7.3.4 DCON_Write_DO ()

- **Description:**
  This function sends a group of digital output data to the PDS series module.

- **Syntax:**
  DCON_Write_DO(unsigned char **cComPort**, short **iAddress**, short **iSlot**, short **iDO_TotalCh**, unsigned long **lDO_Value**, short **iCheckSum**, short **iTimeOut**);

- **Parameters:**

cComPort	COM Port number, 1 ~ 255
iAddress	Module address, **1** for the PDS series module
iSlot	**–1** for the PDS series module
iDO_TotalCh	total DO channel count on the PDS series module
lDO_Value	digital output data
iCheckSum	0: disabled or 1: enabled
iTimeout	timeout setting, default = 100 (unit: ms)

- **Return:**
  0 → no error; **Others** → error codes

7.3.5 DCON_Write_DO_Bit ()

- **Description:**
  This function sends one bit of digital output data to the PDS series module.

- **Syntax:**
  DCON_Write_DO_Bit(unsigned char **cComPort**, short **iAddress**, short **iSlot**, short **iChannel**, short **iDO_TotalCh**, short **iBitValue**, short **iCheckSum**, short **iTimeOut**);

- **Parameters:**

cComPort	COM Port number, 1 ~ 255
iAddress	Module address, **1** for the PDS series module
iSlot	**–1** for the PDS series module
iChannel	The digital output channel No.
iDO_TotalCh	total DO channel count on the PDS series module
iBitValue	**1** bit of digital output data, 0 = off, 1 = on
iCheckSum	0: disabled or 1: enabled
iTimeout	timeout setting, normal = 100, unit: ms

- **Return:**
  0 → no error; **Others** → error codes

7.3.6 DCON_Read_DIO ()

- **Description:**

  This function reads the DO and DI lines status.

- **Syntax:**

  DCON_Read_DIO(unsigned char **cComPort**, short **iAddress**, short **iSlot**, short **iDI_TotalCh**, short **iDO_TotalCh**, short **iCheckSum**, short **iTimeOut**, unsigned long ***lDI_Value**, unsigned long ***lDO_Value**, char ***cDI_BitValue**, char ***cDO_BitValue**);

- **Input Parameter:**

cComPort	COM Port number, 1 ~ 255
iAddress	Module address, **1** for the PDS series module
iSlot	**–1** for the PDS series module
iDI_TotalCh	total DI channel count on the PDS series module
iDO_TotalCh	total DO channel count on the PDS series module
iCheckSum	0: disabled or 1: enabled
iTimeout	Timeout setting, normal = 100, unit: ms
iDI_Value	read digital input data
iDO_Value	read digital output data
cDI_BitValue	read digital input data, Boolean array format
cDO_BitValue	read digital output data, Boolean array format

- **Return:**

  0 → no error

  Others → error codes

8. Virtual I/O Commands

- Command Format: **(Leading)(Address)(Command)[CHK](cr)**
- Response Format: **(Leading)(Address)(Data)[CHK](cr)**

(Address)	2-character, “**01**” for PDS virtual I/O
[CHK]	2-character checksum, no checksum for PDS virtual I/O
(cr)	carriage return (0x0D) for ending character of command

- **Checksum Calculation:**
  1. Calculate the ASCII sum of all characters in the command (or response) string except for the return character (cr).
  2. Mask the sum of the string with 0ffh

- **Example:**

Command string: $012 (cr)
Sum of the string = ‘$’ + ‘0’ + ‘1’ + ‘2’
= 24 h + 30 h + 31 h + 32 h
= B7 h
The checksum is **B7 h**, and **[CHK]** =”**B7**”.

Command string with checksum: $012B7 (cr)

Response string: !01300600 (cr)
Sum of the string = ‘!’ + ‘0’ + ‘1’ + ‘3’ + ‘0’ + ‘0’ + ‘6’ + ‘0’ + ‘0’
= 21 h + 30 h + 31 h + 33 h + 30 h + 30 h + 36 h + 30 h + 30 h
= 1AB h
The checksum is **AB h**, and **[CHK]** =”**AB**”.

Response string with checksum: !01300600AB (cr)

General Command Sets			
Command	**Response**	**Description**	**Section**
$AA5	!AAS	Reads the Reset Status	8.1
$AA6	!AA(Data)	Reads the Digital I/O Status	8.2
$AAC	!AA	Clears the Latched Digital Input	8.3
$AACn	!AA	Clears the Digital Input Count	8.4
$AAGCN	>AA(Data)	Retrieves the I/O Channel Count	8.5
$AALs	!(Data)	Reads the Latched DI	8.6
$AAF	!AA(Data)	Reads the Firmware Version	8.7
$AAM	!AA(Data)	Reads the Module Name	8.8
@AA	>(Data)	Reads the Digital Input/Output Status	8.9
@AA(Data)	>	Sets the Digital Output	8.10
#AAn	!AA(Data)	Reads the DI counter	8.11
#AA00dd	>	Sets the Multi-channel Output	8.12
#AA1ndd	>	Sets the Single Channel Output	8.13

Host Watchdog Command Sets			
Command	**Response**	**Description**	**Section**
~**	No Reponse	Host is OK	8.14
~AA0	!AASS	Reads the Module Status	8.15
~AA1	!AA	Reset Module Status	8.16
~AA2	!AAeff	Reads the Host Watchdog Timeout Value	8.17
~AA3eff	!AA	Sets the Host Watchdog Timeout Value	8.18
~AA4P	!AA(Data)	Reads the Power-on Value for D/O	8.19
~AA4S	!AA(Data)	Reads the Safe Value for the D/O	8.20
~AA5P	!AA	Sets the Power-on Value for the D/O	8.21
~AA5S	!AA	Sets the Safe Value for the D/O	8.22

Note: All commands require a carriage return (0x0D) for ending character.

8.1 $AA5

- **Description**:

This function reads the reset status

- **Syntax**:

$AA5[CHK](cr)

$	A delimiter character
AA	The address of the module (01 only for PDS)
5	A command for reading the status

- **Response**:

Valid Command: **!AAs[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*There will be no response if there is a syntax error or a communication error.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
s	The reset status: 1 = The Module has been reset, and the status was cleared to 0 after using this command. 0 = the module has never been reset

- **Example**:

Command	**Response**
$015	!011
1. Reads the reset status. Returns read for the first time	
$015	!010
1. Reads the reset status. Returns that no reset has occurred	

- **Related Topics**:

Sec. 8.16 ~AA1

8.2 $AAG

- **Description:**

This function reads the status of the digital I/O channels

- **Syntax:**

$AA6[CHK](cr)

$	A delimiter character
AA	The address of the module (01 only for PDS)
6	A command for reading the digital I/O status

- **Response:**

Valid Command: **!ddff00[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A Syntax error or a communication error will result in no response.

!	A delimiter for valid command
?	A delimiter for invalid command
AA	The address of the module (01 only for PDS)
dd	The current status of the digital output channels
ff	The status of the digital input channel

- **Example:**

Command	Response
$016	!0F0000

1. Reads the digital input/output status. Returns 0F00;
2. The status of digital output channels 3 to 0 is set as on.
3. The status of all digital input channels is set to off.

- **Related Commands:**

Sec. 8.9 @AA

8.3 $AAC

- **Description:**

This function clears the latched status of the digital Input

- **Syntax:**

$AAC[CHK](cr)

$	A delimiter character
AA	The address of the module (01 only for PDS)
C	A command for clearing latched digital inputs

- **Response:**

Valid Command: **!AA[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*Syntax error or a communication error will result in no response.

!	A delimiter for valid command
?	A delimiter for invalid command
AA	The address of the module (01 only for PDS)

- **Example:**

Command	**Response**
$01L0	!FFFF00
1. Reads the latch-low data. Returns FFFF.	
$01C	!01
1. Clears the latched digital inputs. Returns success.	
$01L0	!000000
1. Reads the latch-low data. Returns 0000.	

- **Related Commands:**

Sec. 8.6 $AALs

8.4 $AACn

- **Description:**

This function clears the digital input counter

- **Syntax:**

$AACn[CHK](cr)

$	A delimiter character
AA	The address of the module (01 only for PDS)
C	A command for clearing the digital input count
n	The digital input channel number

- **Response:**

Valid Command: **!AA[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)

- **Example:**

Command	**Response**
#010	!0100123
1. Reads the counter value on digital input channel 0. Returns 123.	
$01C0	!01
1. Clears the counter value on digital input channel 0. Returns success.	
#010	!0100000
1. Reads the counter value on digital input channel 0. Returns 0.	

- **Related Commands:**

Sec. 8.11 #AAn

8.5 $AAGCN

- **Description:**

This function reads the digital input/output channel count

- **Syntax:**

$AAGCN[CHK](cr)

$	A delimiter character
AA	The address of the module (01 only for PDS)
GCN	A command for reading the digital input/output channel count

- **Response:**

Valid Command: **>DINxxDONxx[CHK](cr)**

Valid Command: **>DONxx[CHK](cr)** (DO only)

Invalid Command: **?AA[CHK](cr)**

* A Syntax error or a communication error will result in no response.

>	A delimiter for valid command
?	A delimiter for invalid command
AA	The address of the module (01 only for PDS)
DINxx	DIN: The digital input channel xx : The total channel count
DONxx	DON: The digital output channel xx : The total channel count

- **Example:**

Command	**Response**
$01GCN	>DIN01DON02
1. Reads the total I/O channel count for the module. Returns that the module is equipped with 1 digital input channel and 2 digital output channels.	

8.6 $AALs

- **Description:**

This function reads the latched digital Input

- **Syntax:**

$AALs[CHK](cr)

$	A delimiter character
AA	The address of the module (01 only for PDS)
L	A command for reading the latched digital input data
s	1 = read latch-high data 0 = read latch-low data

- **Response:**

Valid Command: **!(Data)[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
(Data)	The read data 1 = the input channel is latched 0 = the input channel is not latched

- **Example:**

Command	**Response**
$01L1	!FF0000
Read the latch-high data. Returns FF00 meaning that, channel 7 to 0 are all latched. (For modules with a DI channel count between 5 and 8)	
$01L1	!F00000
1. Reads the latch-high data. Returns FF00 meaning that, channels 3 to 0 are all latched. (For modules with a DI channel count between 1 and 4)	

- **Related Commands:**

Sec.8.3 $AAC

8.7 $AAF

- **Description:**

This function reads the firmware version details

- **Syntax:**

$AAF[CHK](cr)

$	A delimiter character
AA	The address of the module (01 only for PDS)
F	A command for reading the firmware version details

- **Response:**

Valid Command: **!AA(Data)[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A Syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
(Data)	The firmware version information

- **Example:**

Command	**Response**
$01F	!01A1.00
1. Read the firmware version details. Returns version No. A1.00	

8.8 $AAM

- **Description:**

This function reads the module name

- **Syntax:**

$AAM[CHK](cr)

$	A delimiter character
AA	The address of the module (01 only for PDS)
M	A command for reading the module name

- **Response:**

Valid Command: **!AA(Data)[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
(Data)	The name of module

- **Example:**

Command	**Response**
$01M	!01PDS-721
1. Reads the module name. Returns the name PDS-721	

8.9 @AA

- **Description:**

This function reads the status of the digital input/output

- **Syntax:**

@AA[CHK](cr)

@	A delimiter character
AA	The address of the module (01 only for PDS)

- **Response:**

Valid Command: **>(Data)[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

>	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
(Data)	The status of the DIO

- **Example:**

Command	**Response**
@01	>050F

1. Read the status of the DIO. Returns 050F.
2. The first two bytes indicate the status of the DO. 05 means that channel 0 and channel 2 are 1, and the other channels are 0.
3. The last two bytes indicate the status of the DI. 0F means that all 4 channels read are 1.

8.10 @AA(Data)

- **Description**:

  This function sets the digital output

- **Syntax:**

  @AA(Data)[CHK](cr)

@	A delimiter character
AA	The address of the module (01 only for PDS)
(Data)	The output value

1. (Data) is one character for output channels less than 4
   For the PDS(M)-762(D) modules, this will be from 0 to 3
   For the PDS(M)-732(D), 734(D), 743(D) modules, this will be from 0 to F

2. (Data) is two characters for output channels less than 8
   For the PDS(M)-721(D) modules, this will be from 00 to 7F

- **Response:**

  Valid Command: **>[CHK](cr)**

  Invalid Command: **?[CHK](cr)**

  Ignore Command: **![CHK](cr)**

  *A syntax error or a communication error will result in no response.

>	A delimiter for a valid command
?	A delimiter for an invalid command
!	A delimiter for an ignore command, meaning that the module is in Host Watchdog Timeout Mode, and the output is set to safe value.

- **Example:**

Command	**Response**
@013	>
Output value 3, Returns success	
@011F	!
Output value 1F. Return that the module is in Host Watchdog Timeout Mode and the output command has been ignored.	

8.11 #AAn

- **Description:**

This function reads digital input counter from channel n

- **Syntax**:

#AAn[CHK](cr)

#	A delimiter character
AA	The address of the module (01 only for PDS)
n	The digital channel number (from 0)

- **Response:**

Valid Command: **!AA(Data)[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
(Data)	The digital input counter value in decimal format from 00000 to 65535

- **Example:**

Command	**Response**
#012	!0100103
Read the digital input counter of channel 2. Returns the value 103	
#013	?01
Read the digital input counter of channel 3. Returns an error including that the channel is not available	

- **Related Command:**

Sec. 8.4 $AACn

8.12 #AA00dd

- **Description:**

This function sets the multi-channel Output

- **Syntax:**

#AA00dd[CHK](cr)

#	A delimiter character
AA	The address of the module (01 only for PDS)
00	A command used to set multi-channel output
dd	The output value

- **Response:**

Valid Command: **>[CHK](cr)**

Invalid Command: **?[CHK](cr)**

Ignored Command: **![CHK](cr)**

*A syntax error or a communication error will result in no response.

>	A delimiter for a valid command
?	A delimiter for an invalid command
!	A delimiter for an ignore command, meaning that the module is in Host Watchdog Timeout Mode, and the output is set to safe values

- **Example:**

Command	**Response**
#01000F	>
Set the digital output of channel 3 to 0 as on. Returns success.	
#010005	!
Set the digital output of channel 0 and 2 as on. Returns that the module is in Host Watchdog Timeout Mode, and the output was been set to safe values.	

- **Related Commands:**

Sec. 8.10 @AA(Data), Sec.8.15 ~AA0, Sec.8.16 ~AA1

8.13 #AA1ndd

- **Description:**

This function sets the output of a single channel

- **Syntax:**

#AA1ndd[CHK](cr)

#	A delimiter character
AA	The address of the module (01 only for PDS)
1n	The command used to set the output of a single channel. n is the digital output channel number.
dd	00: sets the digital output channel to off 01: sets the digital output channel to on

- **Response:**

Valid Command: **>[CHK](cr)**

Invalid Command: **?[CHK](cr)**

Ignored Command: **![CHK](cr)**

*A syntax error or a communication error will result in no response.

>	A delimiter for a valid command
?	A delimiter for an invalid command
!	A delimiter for an ignore command, meaning that the module is in Host Watchdog Timeout Mode, and the output is set to safe values

- **Example:**

Command	**Response**
#011201	>
Set the digital output of channel 2 to on. Returns success.	

- **Related Commands:**

Sec. 8.10 @AA(Data), Sec.8.15 ~AA0, Sec.8.16 ~AA1

8.14 ~**

- **Description**:

  This function is used to let other modules know that the Host is OK

- **Syntax**:

  ~**[CHK](cr)

~	A delimiter character
**	The command for all modules

- **Response**:

  No response

- **Example**:

Command	Response
~**	No response

- **Related Commands**:

  Sec.8.15 ~AA0, Sec.8.16 ~AA1, Sec.8.17 ~AA2, Sec8.18 ~AA3eff, Sec.8.19 ~AA4P, Sec.8.20 ~AA4S, Sec.8.21 ~AA5P, Sec.8.22 ~AA5S

8.15 ~AA0

- **Description**:

The function reads the status of the Host Watchdog

- **Syntax**:

~AA0[CHK](cr)

~	A delimiter character
AA	The address of the module (01 only for PDS)
0	The command used to read the status of the module

- **Response**:

Valid Command: **!AASS[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
SS	The status of the module 00 = The Host watchdog status has been cleared, or no timeout occurred. 04 = The Host Watchdog has been set, timeout occurred. Use command ~AA1 to clear the status that recorded in EEPROM.

- **Example**:

See the ~AA3eff example in Sec 8.18

- **Related Commands**:

Sec.8.15 ~AA0, Sec.8.16 ~AA1, Sec.8.17 ~AA2, Sec.8.18 ~AA3eff, Sec.8.19 ~AA4P, Sec.8.20 ~AA4S, Sec.8.21 ~AA5P, Sec.8.22 ~AA5S

8.16 ~AA1

- **Description:**

This function clears the status of the Host Watchdog

- **Syntax:**

~AA1[CHK](cr)

~	A delimiter character
AA	The address of the module (01 only for PDS)
1	The command used to reset the status of the module

- **Response:**

Valid Command: **!AA[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)

- **Example:**

See the ~AA3eff example in Sec. 8.18

- **Related Commands:**

Sec. 8.15 ~AA0, Sec. 8.16 ~AA1, Sec. 8.17 ~AA2, Sec. 8.18 ~AA3eff, Sec. 8.19 ~AA4P, Sec. 8.20 ~AA4S, Sec. 8.21 ~AA5P, Sec. 8.22 ~AA5S

8.17 ~AA2

- **Description**:

  This function reads the Host Watchdog Timeout Value

- **Syntax**:

  ~AA2[CHK](cr)

~	A delimiter character
AA	The address of the module (01 only for PDS)
2	The command used to read the Host Watchdog Timeout Value

- **Response**:

  Valid Command: **!AAeff[CHK](cr)**

  Invalid Command: **?AA[CHK](cr)**

  *A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
e	The status of the Host Watchdog, 1 = Enabled, 0 = Disabled
ff	The timeout value in Hex format. The unit is 0.1 seconds 01 = 0.1 seconds FF = 25.5 seconds

- **Example**:

  See the ~AA3eff example in Sec. 8.18

- **Related Commands**:

  Sec. 8.15 ~AA0, Sec. 8.16 ~AA1, Sec. 8.17 ~AA2, Sec. 8.18 ~AA3eff, Sec. 8.19 ~AA4P, Sec. 8.20 ~AA4S, Sec. 8.21 ~AA5P, Sec. 8.22 ~AA5S

8.18 ~AA3eff

- **Description:**

This function sets the Host Watchdog Timeout Value

- **Syntax:**

~AA3eff[CHK](cr)

~	A delimiter character
AA	The address of the module (01 only for PDS)
3	The command used to set the Host Watchdog Timeout Value
e	1 = Enabled, 0 = Disabled
ff	The timeout value, from 01 to FF, the unit is 0.1 second

- **Response:**

Valid Command: **!AA[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)

- **Example:**

Command	**Response**
~010	!0100
Read the status of the module. Returns the status of the Host Watchdog Timeout has been cleared.	
~013164	!01
Set the status of the Host Watchdog Timeout Value to 10.0 seconds and enable the Host Watchdog. Return Success.	
~012	!01164
Read the Host Watchdog Timeout Value. Returns the Host Watchdog Timeout Value is 10.0 seconds, and the Host Watchdog is enabled.	
~**	No response
If the ~** command is not sent within 10 seconds, the LED on the module will begin to flash. The LED indicates that the status of the Host Watchdog Timeout is set.	

~010	!0104
Read the status of the module. Returns the status of the Host Watchdog Timeout Value is set.	
~012	!01064
Read the Host Watchdog Timeout Value. Returns the Host Watchdog Timeout Value is 10.0 seconds, and the Host Watchdog is disabled.	
~011	!01
Reset the status of the Host Watchdog Timeout. Returns success, and the LED stops flashing.	
~010	!0100
Read the module status. Returns that the status of the Host Watchdog Timeout has been clear.	

- **Related Commands:**

  Sec. 8.15 ~AA0, Sec. 8.16 ~AA1, Sec. 8.17 ~AA2, Sec. 8.18 ~AA3eff, Sec. 8.19 ~AA4P, Sec. 8.20 ~AA4S, Sec. 8.21 ~AA5P, Sec. 8.22 ~AA5S

8.19 ~AA4P

- **Description:**

This function reads the Power-on Values for the DO

- **Syntax:**

~AA4P[CHK](cr)

~	A delimiter character
AA	The address of the module (01 only for PDS)
4P	The command used to read the power-on value for the DO

- **Response:**

Valid Command: **!AA(Data)[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
(Data)	The power-on values

- **Example:**

Command	**Response**
~014P	!01000F
Read the power-on values. Returns the power-on value 0F	

- **Related Commands:**

Sec. 8.21 ~AA5P

8.20 ~AA4S

- **Description**:

  This function reads the Safe Values for the DO

- **Syntax**:

  ~AA4S[CHK](cr)

~	A delimiter character
AA	The address of the module (01 only for PDS)
4S	The command used to read safe values for the DO

- **Response**:

  Valid Command: **!AA(Data)[CHK](cr)**

  Invalid Command: **?AA[CHK](cr)**

  *A Syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)
(Data)	The power-on values

- **Example**:

Command	**Response**
~014S	!01000F
Read the safe values. Returns the power-on value 0F	

- **Related Commands**:

  Sec. 8.22 ~AA5S

8.21 ~AA5P

- **Description:**

This function sets the Power-on Value for the DO

- **Syntax:**

~AA5P[CHK](cr)

~	A delimiter character
AA	The address of the module (01 only for PDS)
5P	The command used to set the power-on value for the DO

- **Response:**

Valid Command: **!AA[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)

- **Example:**

Command	**Response**
@0103	>
Output the value 03. Returns success	
~015P	!01
Set the current output status as power-on value, return success.	

- **Related Commands**:

Sec. 8.19 ~AA4P

8.22 ~AA5S

- **Description**:

This function sets the Safe Value for the DO

- **Syntax**:

~AA5S[CHK](cr)

~	A delimiter character
AA	The address of the module (01 only for PDS)
5S	The command used to set the safe value for the DO

- **Response**:

Valid Command: **!AA[CHK](cr)**

Invalid Command: **?AA[CHK](cr)**

*A syntax error or a communication error will result in no response.

!	A delimiter for a valid command
?	A delimiter for an invalid command
AA	The address of the module (01 only for PDS)

- **Example**:

Command	**Response**
@0103	>
Output the value 03. Returns success	
~015S	!01
Set the current output status as the safe value. Returns success.	

- **Related Commands**:

Sec. 8.20 ~AA4S

8.23 Application Notes

■ Hot Watchdog Status

The Power On Reset or Module Watchdog Reset will return all output settings to Power On Values. The Host Watchdog Timeout will return all output values to Safe Values. Additionally, the status of the Host Watchdog, which can be read using the ~AA0 command, will be recorded as 04, and output commands will be ignored until the status is cleared to 0 by using command ~AA1.

■ Dual Watchdog Operation

Dual Watchdog = Module Watchdog + Host Watchdog

The Module Watchdog is a hardware reset circuit that is used to monitor the operating status of the module. When working in harsh or noisy environments, the module may be interrupted by external signals. The circuit can let the module reboot automatically and to work continues without halt.

The Host Watchdog is a software function that is used to monitor the operating status of the host. Its purpose is to keep PDS's output in a safe state when there is network communication problem or host PC halt. If the timeout interval expires, the module will turn all outputs to the predefined Safe Value and prevent unexpected situation from occurring.

The PDS series with Dual Watchdog will make the control system more reliable and stable.

■ Reset Status

The Reset Status is set when the module is powered on or if it is reset by the Module Watchdog. The status can be cleared by using the Reset Status ($AA5) command. This is useful for checking the operating status of the module. When the Reset Status is set it means that the module has been reset and the output can be changed to the Power On Value. When the Reset Status is clear it means the module was not been reset, and the output was not changed.

■ Digital Output

The output status of the module has three different conditions:

1. **Safe Value**: If the Host Watchdog Timeout status is set, the output is set to the Safe Value. If the module receives an output command, such as @AA(Data) or #AABBDD, the module will ignore the command and return '!', and will not change the output to the output command value. The Host Watchdog Timeout status is set and store into EEPROM while the Host Watchdog Timeout interval has expired, and can only be cleared by setting the command ~AA1.

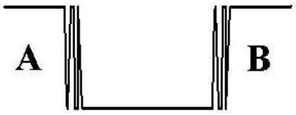
   If user needs to change the output, the Host Watchdog Timeout status needs to be cleared first, and an output command need to be sent to change the output to the desired value.

2. **PowerOn Value:** Once the module is reset, and the Host Watchdog Timeout status has been cleared, the output of the module is set to a predefined PowerOn Value.

3. **Output command value:** If the Host Watchdog Timeout status is clear, and the user issues a digital output command, to the module to change the output value, such as @AA (Data) or #AABBDD, the module will respond "success" (receive >).

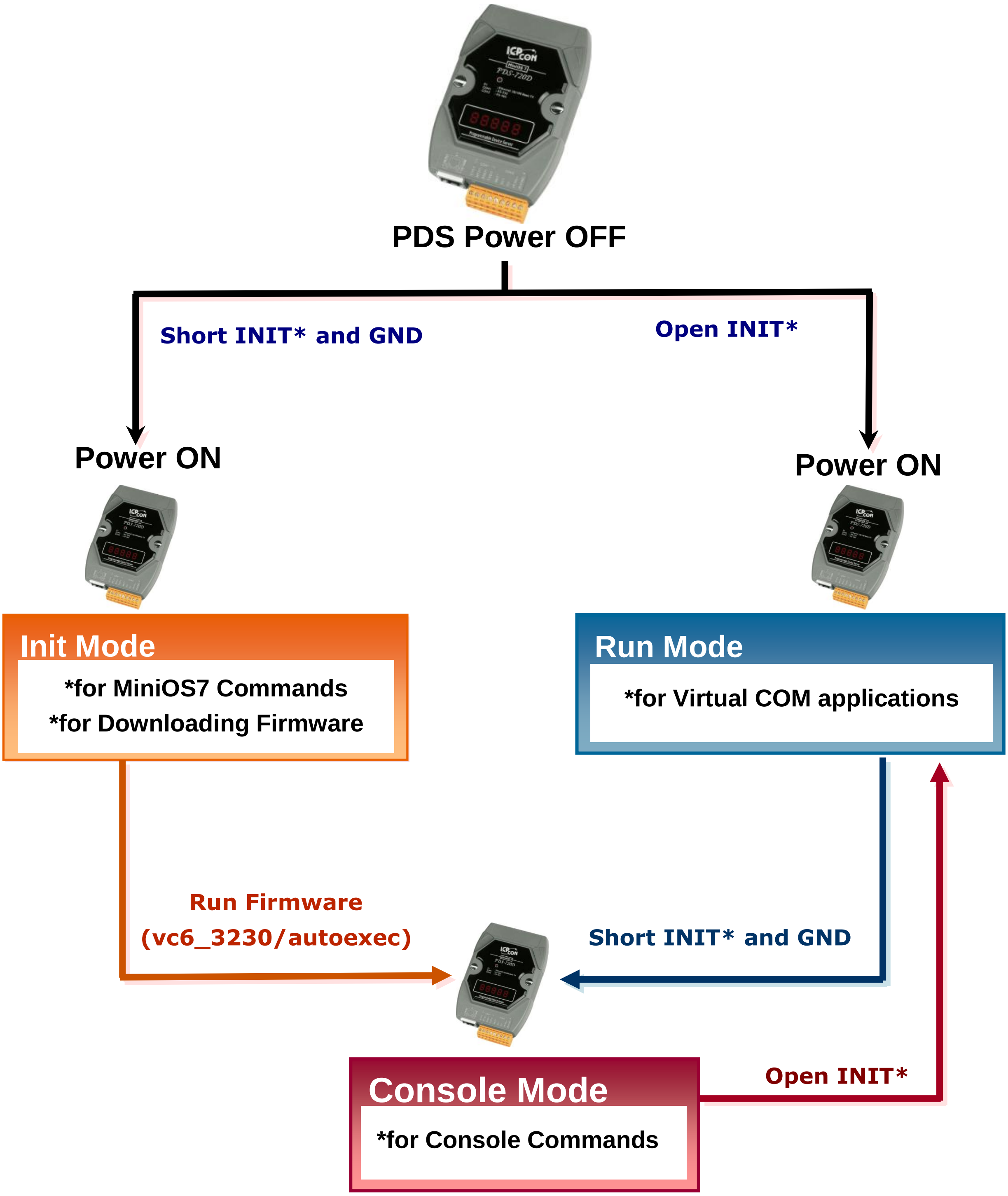
■ Latch Digital Input

If, for example, the user connects a key switch to the digital input channel and wants to read the keystrokes, the key input is a digital input pulse, and the keystroke will be lost. By using command $AA6 to read the A and B positions, the response would be that there was no keystroke and the keystroke information will be lost. The read latch-low digital input command, $AAL0, will solve this problem. When issuing a $AAL0 command at the A and B positions, the response will denote that there is a low pulse between A and B position signifying a keystroke.



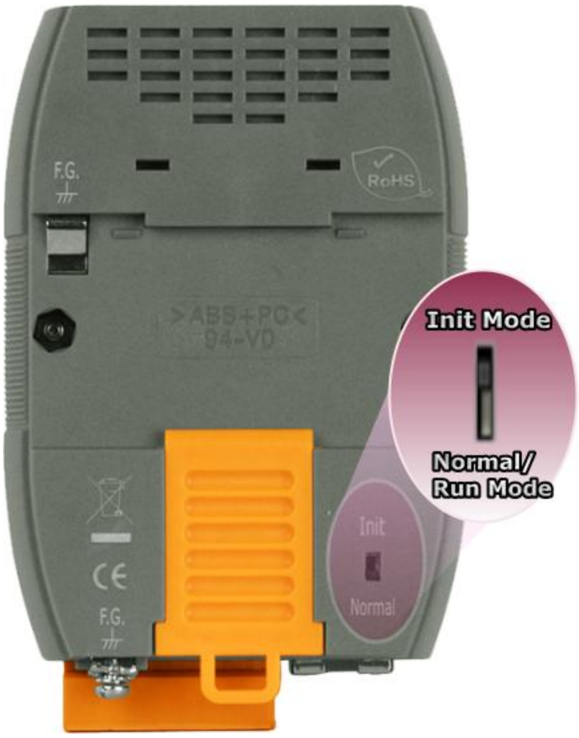
9. Console / Telnet Commands List

9.1 Operation Flowchart

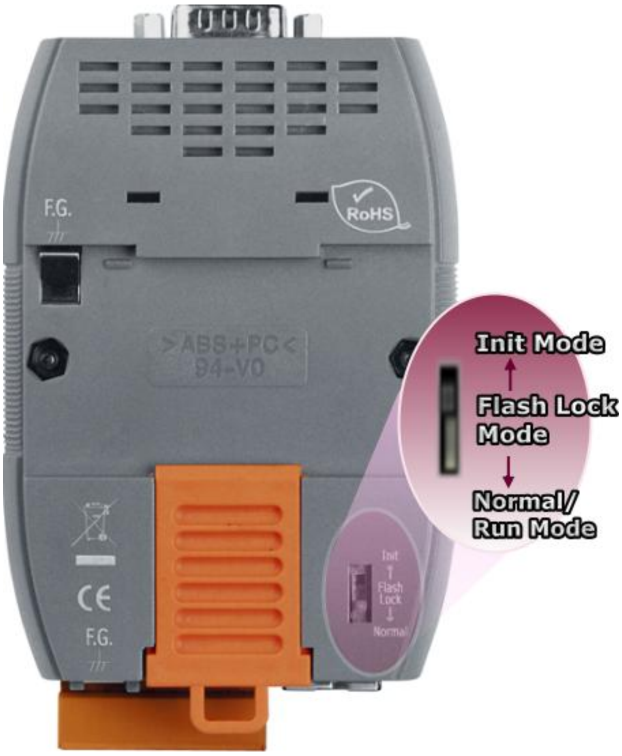


9.2 Regulate Init/Normal Mode

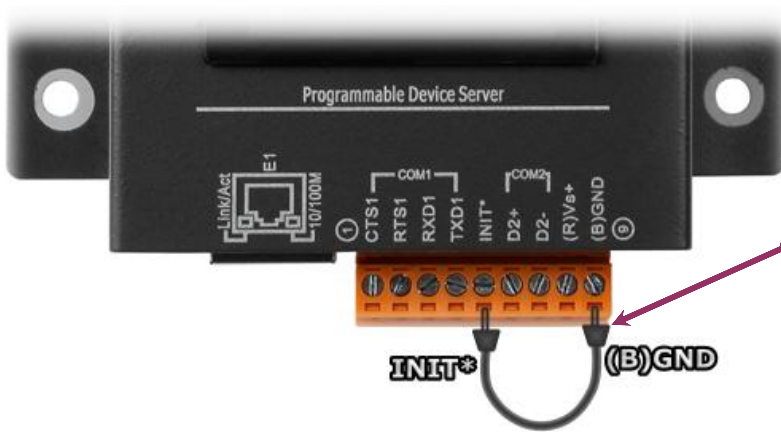
- **PDS-700(D), PPDS-700(D)-MTCP**



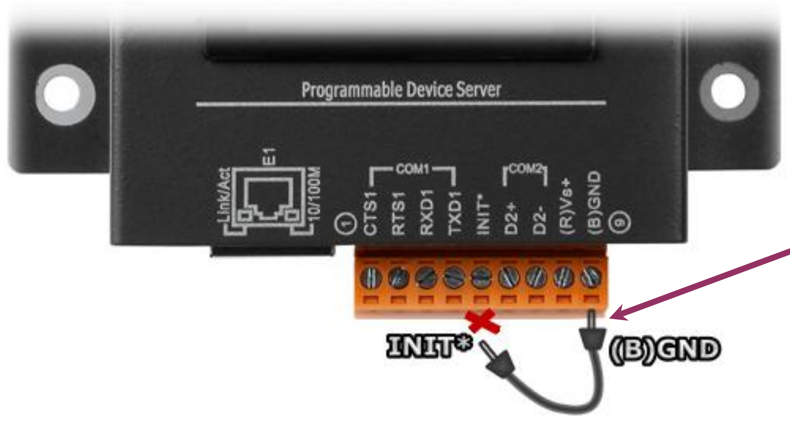
- **DS-700, PDS-782(D)-25**



■ PDSM-700(D), PPDSM-700(D)-MTCP

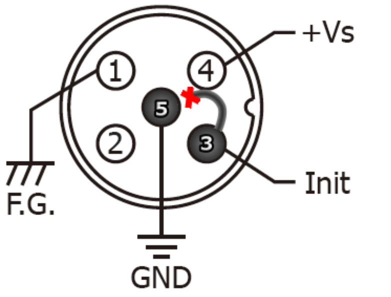

Programmable Device Server
Link/Act
E1
10/100M
COM1
COM2
① CTS1 RTS1 RXD1 TXD1 INIT* D2+ D2- (R)Vs+ (B)GND ⑨
INIT*
(B)GND

Init Mode:
Short INIT* and GND


Programmable Device Server
Link/Act
E1
10/100M
COM1
COM2
① CTS1 RTS1 RXD1 TXD1 INIT* D2+ D2- (R)Vs+ (B)GND ⑨
INIT*
(B)GND

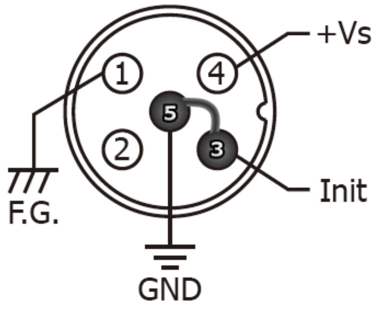
Run Mode:
Open INIT*

■ PPDS-700-IP67

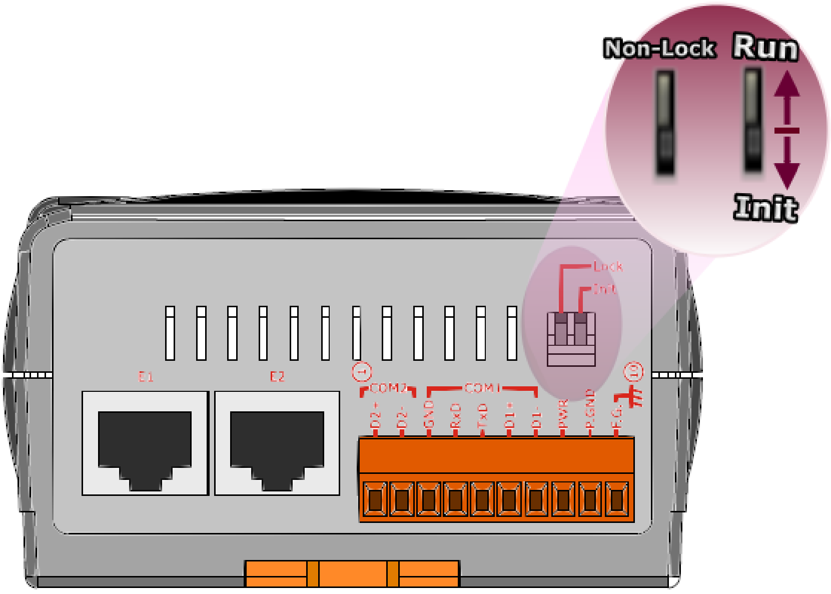


Run Mode: Open INIT*

Init Mode: Short INIT* and GND



- **PDS-5105D-MTCP**



9.3 Comparison Sheet (Init/Run/Console Modes)

Mode	Firmware	Init* and GND pins	VCOM Commands	Telnet Commands	Console Commands
Init	*Stop*	-	*No*	*No*	*No*
	Init Mode is used to upgrade firmware and accepts MiniOS7 commands (from PDS.COM1) only.				
Run	*Running*	*Open*	***Yes***	***Yes***	*No*
	Run Mode is used for Virtual COM applications, and accepts Virtual COM commands (TCP port 10000) and Telnet commands (TCP port 23).				
Console	*Running*	*Short*	***Yes***	***Yes***	***Yes***
	Console Mode is used to configure the Virtual COM. PDS.COM1 is the console port that accepts console commands while other ports are still working with Virtual COM applications.				

9.4 Command List

Sec.	Command	Description
9.4.1	IPFILTER	Retrieves/Sets the IP addresses that are allowed to access the PDS.
9.4.2	IPCONF	Queries the network configuration. (IP/Mask/Gateway/MAC addresses).
9.4.3	SOCKET	Lists all the status of sockets (Listen/Not Used Yet) together with the type of each socket (TCP Server: Port No./UDP/Unused).
9.4.4	COM	Queries or sets the configuration of the COM Ports (Baud Rate/Parity /Stop Bits)
9.4.5	Broadcast	Queries or sets the Broadcast parameter, which determines whether or not the module can receive Broadcast packets.
9.4.6	SystemTimeout	If PDS has no network communications during the SystemTimeout period, the PDS will reboot it-self automatically.
9.4.7	SocketTimeout	If there is no data send/receive on the connection during the SocketTimeout period, the PDS will close the connection automatically.
9.4.8	M	Gets/Sets the echo mode. /M0: Transparent Mode, Multi-Echo, Data-Shared. /M1: Slave Mode, Single-Echo, None-Shared.
9.4.9	EchoCmdNo	Queries or sets the EchoCmdNo parameter that enables or disables adding Command Number before response.
9.4.10	EndChar	Sets a character that determines the end of a response string.
9.4.11	IP	Queries or sets the IP address.
9.4.12	MASK	Queries or sets the subnet Mask value.
9.4.13	GATEWAY	Queries or sets the Gateway address.
9.4.14	MAC	Queries the MAC address.
9.4.15	NAME	Queries the module name.
9.4.16	ALIAS	Sets the alias for a PDS.
9.4.17	DHCP	Enables/Disables the DHCP client.
9.4.18	UDP	Sets whether to reply to a UDP search command.
9.4.19	VER	Queries the version information
9.4.20	SAVE	Determines whether or not backup copies of the "autoexec.bat" and "vcom.ini" files are saved when using the “load” command.
9.4.21	LOAD	Loads file to the built-in flash disk on PDS. It should be used to update firmware only.
9.4.22	CONFIG	Restores the factory default settings.
9.4.23	RESET	Reboots the PDS module.
9.4.24	QUIT	Exits the running firmware.

9.4.1 IPFILTER

Description: This command is used to query or edit IP filter table. The IP filter table restricts the access of packets based on the IP header. If one or more IP addresses are saved into the IP filter table, only clients whose IP is specified in the IP filter table can access the PDS.

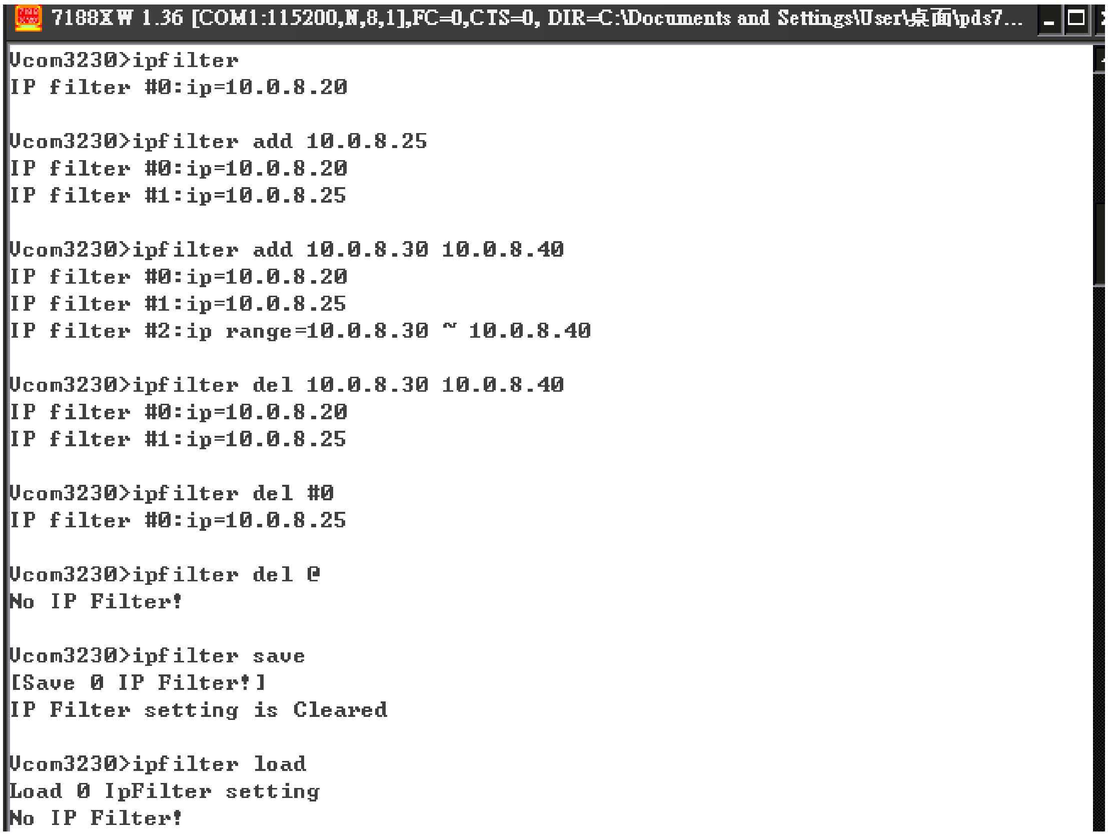
Effect: Immediate

Command	Arguments	Description
ipfilter		Queries the IP filter table.
ipfilter	ADD ip1 ADD ip1 ip2	Adds an IP address to the IP filter table. Adds a range of IP addresses (ip1 ~ ip2) to the IP filter table.
ipfilter	DEL ip1 DEL ip1 ip2	Deletes an IP address (ip1) from the IP filter table. Deletes a range of IP addresses (ip1 ~ ip2) from the IP filter table. The IP address that follows the DEL command should already be listed in the IP filter table.
ipfilter	DEL #n	Deletes item "n" from the IP filter table.
ipfilter	DEL @	Deletes all items from the IP filter table.
ipfilter	SAVE	Saves the IP filter table to the EEPROM. If the IP filter table is empty, the data in EEPROM will be cleared.
ipfilter	LOAD	Loads the IP filter table from the EEPROM.

※ The IP filter table is loaded automatically when the PDS is booted.

※ Use the "ipfilter save" command to save a new IP filter table to the EEPROM.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>ipfilter
IP filter #0:ip=10.0.8.20

Vcom3230>ipfilter add 10.0.8.25
IP filter #0:ip=10.0.8.20
IP filter #1:ip=10.0.8.25

Vcom3230>ipfilter add 10.0.8.30 10.0.8.40
IP filter #0:ip=10.0.8.20
IP filter #1:ip=10.0.8.25
IP filter #2:ip range=10.0.8.30 ~ 10.0.8.40

Vcom3230>ipfilter del 10.0.8.30 10.0.8.40
IP filter #0:ip=10.0.8.20
IP filter #1:ip=10.0.8.25

Vcom3230>ipfilter del #0
IP filter #0:ip=10.0.8.25

Vcom3230>ipfilter del @
No IP Filter!

Vcom3230>ipfilter save
[Save 0 IP Filter!]
IP Filter setting is Cleared

Vcom3230>ipfilter load
Load 0 IpFilter setting
No IP Filter!
```

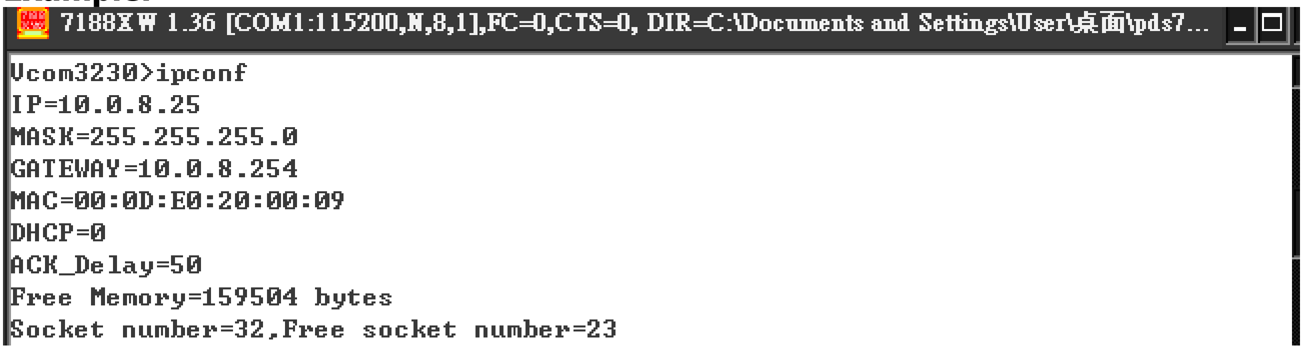
9.4.2 IPCONF

Description: This command is used to display the network configuration information, such as IP/Mask/Gateway/MAC addresses and the status of DHCP/ACK_Delay/Free Memory/Socket status.

Effect: Immediate

Command	Argument	Description
ipconf		Queries the network configuration.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>ipconf
IP=10.0.8.25
MASK=255.255.255.0
GATEWAY=10.0.8.254
MAC=00:0D:E0:20:00:09
DHCP=0
ACK_Delay=50
Free Memory=159504 bytes
Socket number=32,Free socket number=23
```

9.4.3 SOCKET

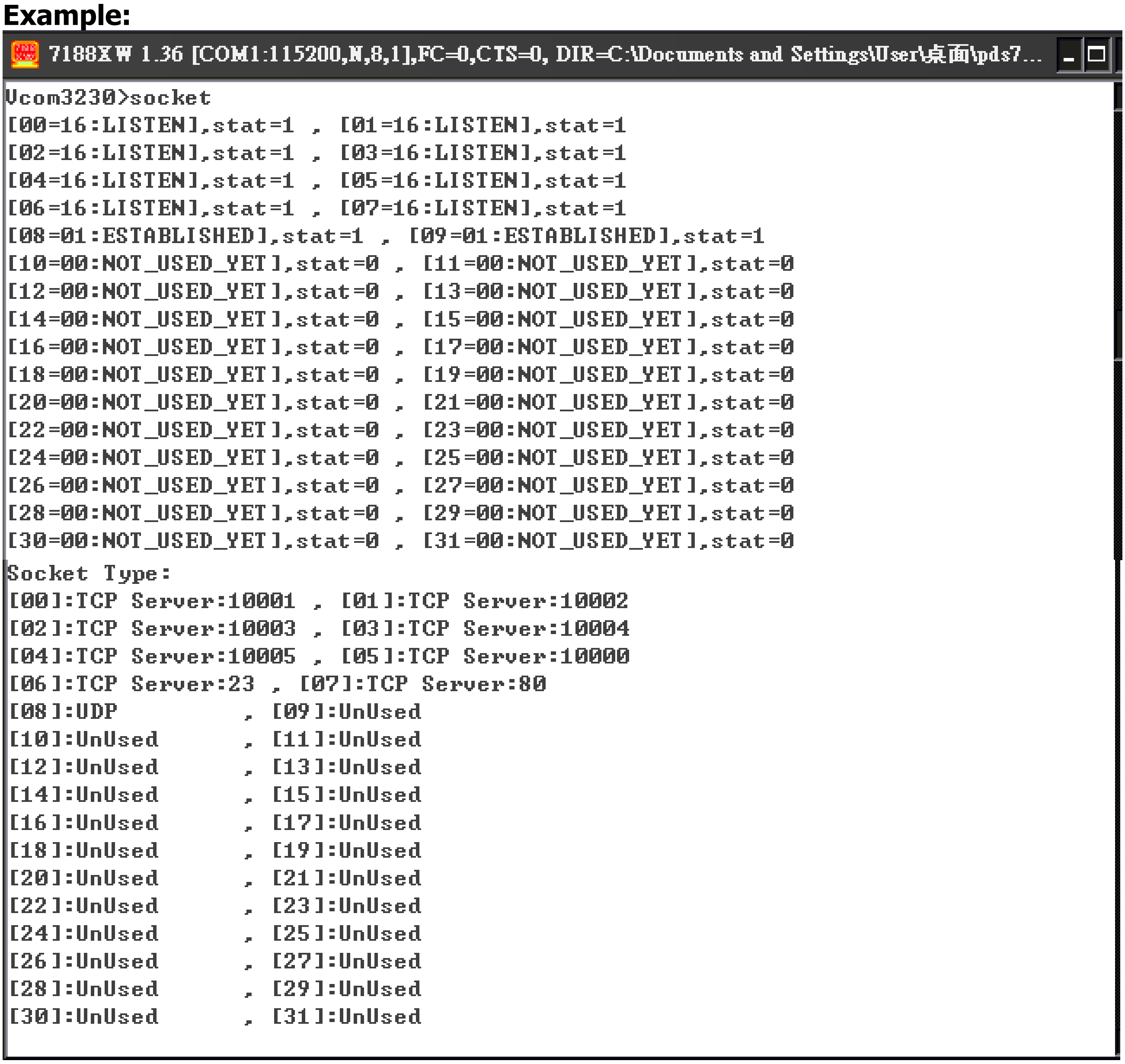
Description: This command lists the status of all sockets (Listen/Not Used Yet) together with the type of each socket (TCP Server: Port No./UDP/Unused)

If stat = 1, the socket is used. If stat = 0, the socket is not yet used.

Take Effect: Immediately

Command	Argument	Description
socket		Lists the status of all sockets.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>socket
[00=16:LISTEN],stat=1 , [01=16:LISTEN],stat=1
[02=16:LISTEN],stat=1 , [03=16:LISTEN],stat=1
[04=16:LISTEN],stat=1 , [05=16:LISTEN],stat=1
[06=16:LISTEN],stat=1 , [07=16:LISTEN],stat=1
[08=01:ESTABLISHED],stat=1 , [09=01:ESTABLISHED],stat=1
[10=00:NOT_USED_YET],stat=0 , [11=00:NOT_USED_YET],stat=0
[12=00:NOT_USED_YET],stat=0 , [13=00:NOT_USED_YET],stat=0
[14=00:NOT_USED_YET],stat=0 , [15=00:NOT_USED_YET],stat=0
[16=00:NOT_USED_YET],stat=0 , [17=00:NOT_USED_YET],stat=0
[18=00:NOT_USED_YET],stat=0 , [19=00:NOT_USED_YET],stat=0
[20=00:NOT_USED_YET],stat=0 , [21=00:NOT_USED_YET],stat=0
[22=00:NOT_USED_YET],stat=0 , [23=00:NOT_USED_YET],stat=0
[24=00:NOT_USED_YET],stat=0 , [25=00:NOT_USED_YET],stat=0
[26=00:NOT_USED_YET],stat=0 , [27=00:NOT_USED_YET],stat=0
[28=00:NOT_USED_YET],stat=0 , [29=00:NOT_USED_YET],stat=0
[30=00:NOT_USED_YET],stat=0 , [31=00:NOT_USED_YET],stat=0
Socket Type:
[00]:TCP Server:10001 , [01]:TCP Server:10002
[02]:TCP Server:10003 , [03]:TCP Server:10004
[04]:TCP Server:10005 , [05]:TCP Server:10000
[06]:TCP Server:23 , [07]:TCP Server:80
[08]:UDP         , [09]:UnUsed
[10]:UnUsed      , [11]:UnUsed
[12]:UnUsed      , [13]:UnUsed
[14]:UnUsed      , [15]:UnUsed
[16]:UnUsed      , [17]:UnUsed
[18]:UnUsed      , [19]:UnUsed
[20]:UnUsed      , [21]:UnUsed
[22]:UnUsed      , [23]:UnUsed
[24]:UnUsed      , [25]:UnUsed
[26]:UnUsed      , [27]:UnUsed
[28]:UnUsed      , [29]:UnUsed
[30]:UnUsed      , [31]:UnUsed
```

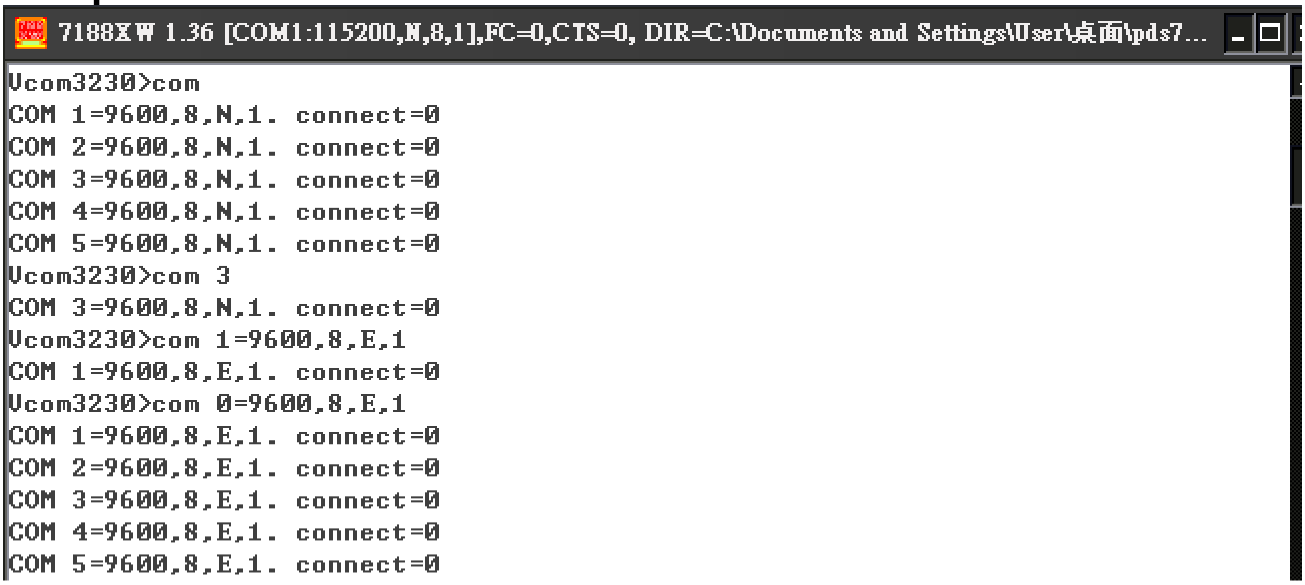
9.4.4 COM

Description: This command queries or sets the configuration of the COM Ports (Baud Rate/Parity/Stop bits).

Effect: Immediate

Command	Arguments	Description
com		Queries the configuration of all COM Ports.
com	n	Queries configuration of COM Port "n". If n = 0, the configuration of all COM Ports will be listed in the same way as using the command "com" above.
com	N = BaudRate, DataBits, Parity,StopBit(s)	Sets the configuration of COM Port "n". If n = 0, the settings will be valid for all Com Ports on the PDS.

Example:



7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>com
COM 1=9600,8,N,1. connect=0
COM 2=9600,8,N,1. connect=0
COM 3=9600,8,N,1. connect=0
COM 4=9600,8,N,1. connect=0
COM 5=9600,8,N,1. connect=0
Vcom3230>com 3
COM 3=9600,8,N,1. connect=0
Vcom3230>com 1=9600,8,E,1
COM 1=9600,8,E,1. connect=0
Vcom3230>com 0=9600,8,E,1
COM 1=9600,8,E,1. connect=0
COM 2=9600,8,E,1. connect=0
COM 3=9600,8,E,1. connect=0
COM 4=9600,8,E,1. connect=0
COM 5=9600,8,E,1. connect=0
```

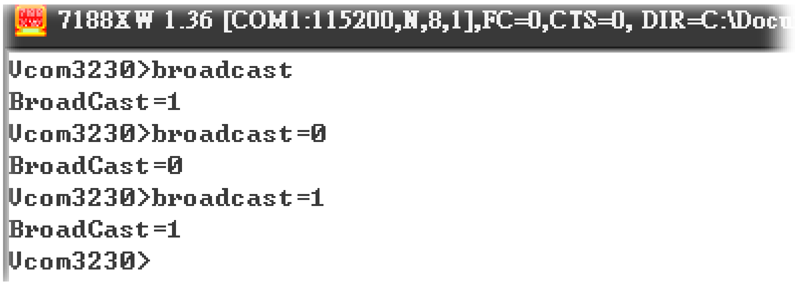
9.4.5 Broadcast

Description: This command is used to Enable/Disable listening broadcast packets on PDS.

Effect: Immediate

Command	Arguments	Description
Broadcast		Queries the Broadcast settings.
Broadcast	= 1	Sets Broadcast = 1. The system is able to receive broadcast packets.
Broadcast	= 0	Sets Broadcast = 0. The system will ignore broadcast packets.

Example:



```
Vcom3230>broadcast
BroadCast=1
Vcom3230>broadcast=0
BroadCast=0
Vcom3230>broadcast=1
BroadCast=1
Vcom3230>
```

9.4.6 SystemTimeout (ms)

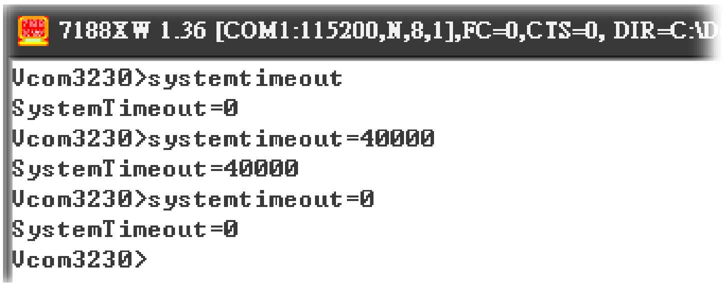
Description: This command queries or sets the system timeout value.

If the SystemTimeout is greater than zero, and the PDS does not receive any packets from any client for longer than the SystemTimeout period, the PDS will reboot itself.

Effect: Immediate

Command	Arguments	Description
SystemTimeout		Queries the SystemTimeout settings.
SystemTimeout	= nnnnn	Sets the SystemTimeout. (Unit : ms) Default factory setting is 300000 ms (= 300 seconds = 5 minutes), min. value is 30000 ms (= 30 seconds)

Example:



```
Vcom3230>systemtimeout
SystemTimeout=0
Vcom3230>systemtimeout=40000
SystemTimeout=40000
Vcom3230>systemtimeout=0
SystemTimeout=0
Vcom3230>
```

9.4.7 SocketTimeout (ms)

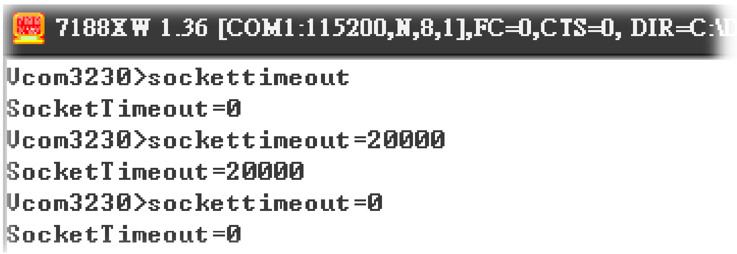
Description: This command is used to query or set the SocketTimeout parameter.

If the SocketTimeout is greater than zero, and the PDS does not receive any data from a client PC for longer than the SocketTimeout period, the PDS will close the socket connection between itself and the client PC.

Effect: Immediate

Command	Arguments	Description
SocketTimeout		Queries the SocketTimeout settings.
SocketTimeout	= nnnnn	Sets the SocketTimeout. (Unit : ms) default = 0 (disable), min = 10000

Example:



```
Vcom3230>sockettimeout
SocketTimeout=0
Vcom3230>sockettimeout=20000
SocketTimeout=20000
Vcom3230>sockettimeout=0
SocketTimeout=0
```

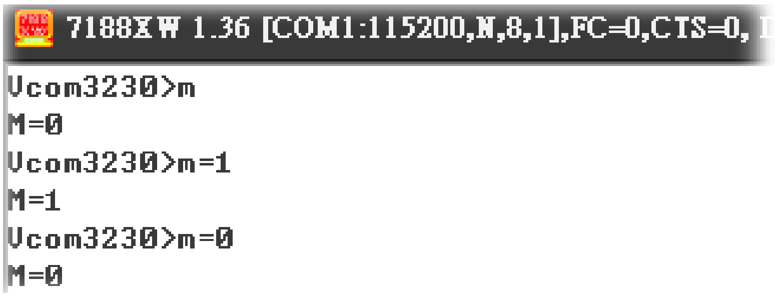
9.4.8 M

Description: This command is used to query or set the echo mode.

Effect: Immediate

Command	Arguments	Description
M		Queries the echo mode settings.
M	= 0	Sets the multi-echo mode to enable. When set to multi-echo mode, the PDS echoes data from a device to all clients that are connected.
M	= 1	Sets the single-echo mode to enable. When set to single-echo mode, the PDS echoes data from a device to the client that requested the service.

Example:



```
Vcom3230>m
M=0
Vcom3230>m=1
M=1
Vcom3230>m=0
M=0
```

9.4.9 EchoCmdNo

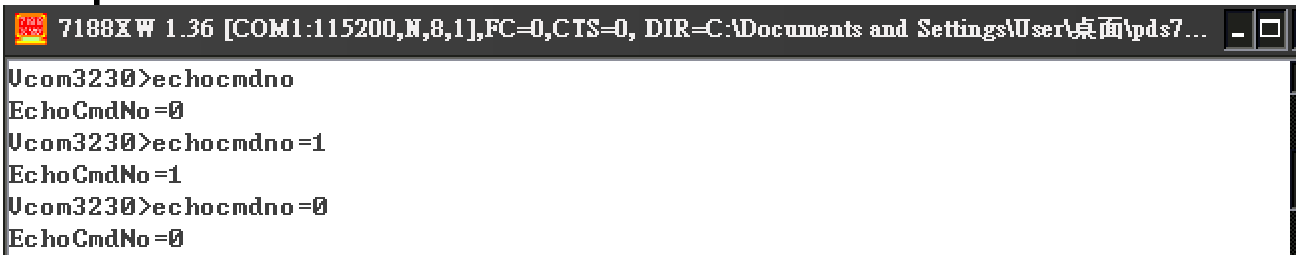
Description: This command is used to query or set the EchoCmdNo parameter.

The EchoCmdNo parameter is used to set whether the PDS prefixes the Virtual COM command to the corresponding response. (Virtual COM commands are used to configure a PDS through TCP port 10000)

Effect: Immediate

Command	Arguments	Description
EchoCmdNo		Queries the EchoCmdNo settings.
EchoCmdNo	= 0	If EchoCmdNo = 0, a Virtual COM command number will not be prefixed to the corresponding response.
EchoCmdNo	= 1	If EchoCmdNo = 1, a Virtual COM command number will be prefixed to the corresponding response.

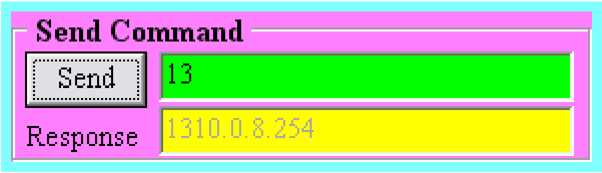
Example:



7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>echocmdno
EchoCmdNo=0
Vcom3230>echocmdno=1
EchoCmdNo=1
Vcom3230>echocmdno=0
EchoCmdNo=0
```

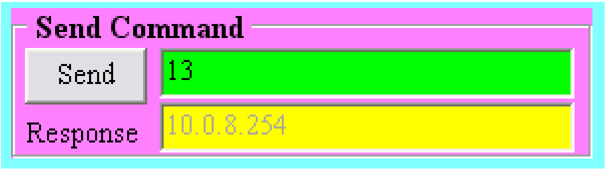
EchoCmdNo = 1



13

1310.0.8.254

EchoCmdNo = 0



13

10.0.8.254

9.4.10 EndChar

Description: This command is used to query or set the EndChar parameter.

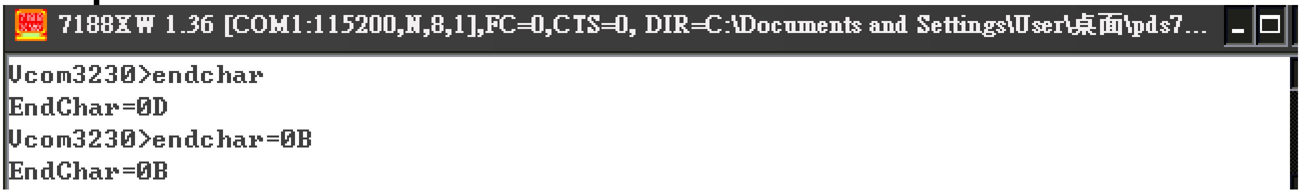
PDS sends out the response string from serial port to TCP client immediately when it received a char on the response string that matching the EndChar.

Set EndChar = 00 to disable the EndChar feature.

Effect: Immediate

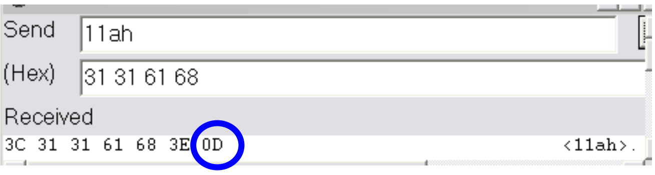
Command	Arguments	Description
Endchar		Queries the endchar setting.
Endchar	= HH	Sets the endchar.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

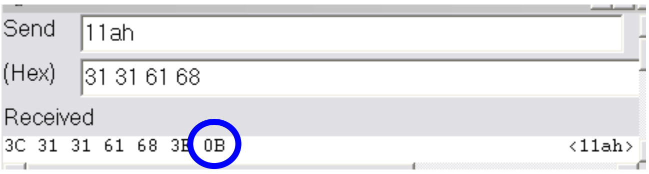
```
Vcom3230>endchar
EndChar=0D
Vcom3230>endchar=0B
EndChar=0B
```

EndChar = 0D



```
Send    11ah
(Hex)   31 31 61 68
Received
3C 31 31 61 68 3E 0D                    <11ah>.
```

EndChar = 0B



```
Send    11ah
(Hex)   31 31 61 68
Received
3C 31 31 61 68 3E 0B                    <11ah>
```

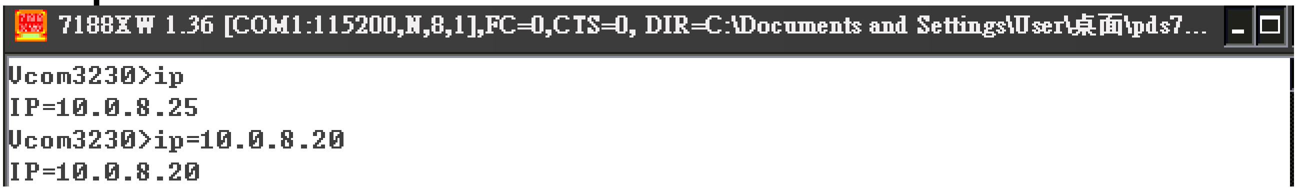
9.4.11 IP

Description: This command is used to query or set the IP address.

Effect: After the next reboot.

Command	Arguments	Description
IP		Queries the IP address.
IP	= xxx.xxx.xxx.xxx	Sets the IP address.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>ip
IP=10.0.8.25
Vcom3230>ip=10.0.8.20
IP=10.0.8.20
```

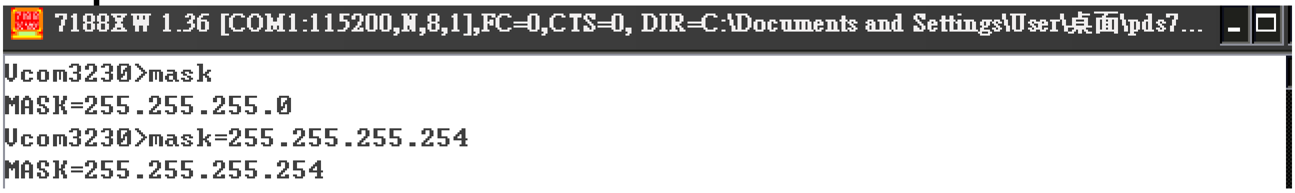
9.4.12 MASK

Description: This command is used to query or set the subnet Mask value.

Effect: After the next reboot.

Command	Arguments	Description
mask		Queries the subnet Mask value.
mask	= xxx.xxx.xxx.xxx	Sets the subnet Mask value.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>mask
MASK=255.255.255.0
Vcom3230>mask=255.255.255.254
MASK=255.255.255.254
```

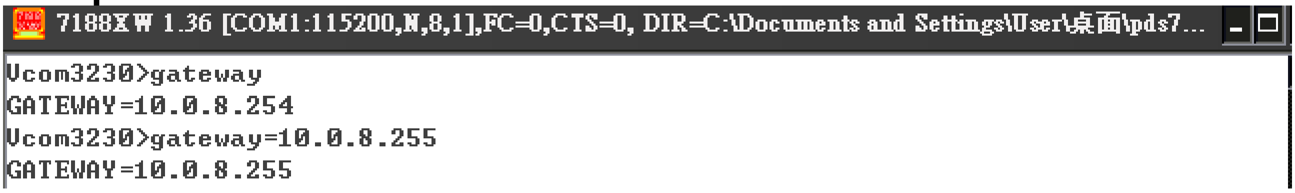
9.4.13 GATEWAY

Description: This command is used to query or set the outgoing Gateway address of the subnet.

Effect: After the next reboot.

Command	Arguments	Description
Gateway		Queries the Gateway address.
Gateway	= xxx.xxx.xxx.xxx	Sets the Gateway address

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>gateway
GATEWAY=10.0.8.254
Vcom3230>gateway=10.0.8.255
GATEWAY=10.0.8.255
```

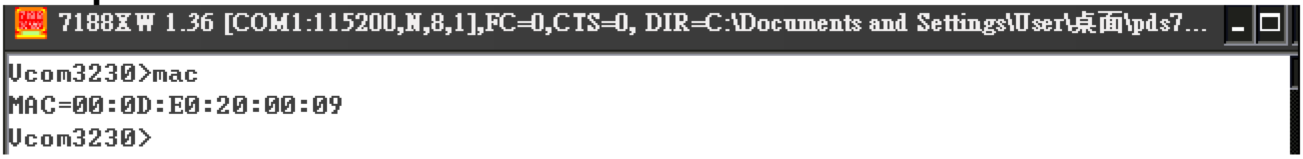
9.4.14 MAC

Description: This command is used to query the MAC address.

Effect: Setting the address is not allowed.

Command	Arguments	Description
Mac		Queries the MAC address.(Setting the address is not allowed)

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>mac
MAC=00:0D:E0:20:00:09
Vcom3230>
```

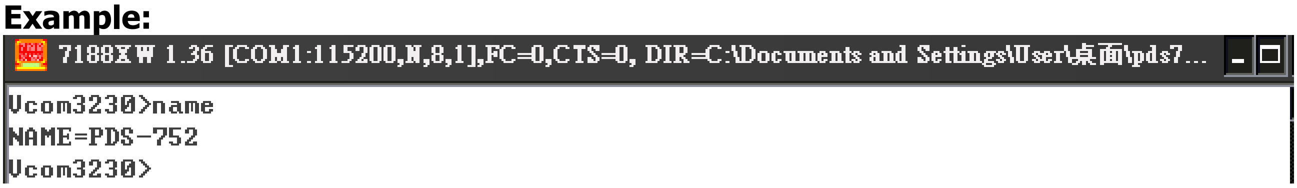
9.4.15 NAME

Description: This command is used to query the name of a PDS module.

Effect: Setting the name is not allowed.

Command	Arguments	Description
name		Queries the name of a PDS module.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>name
NAME=PDS-752
Vcom3230>
```

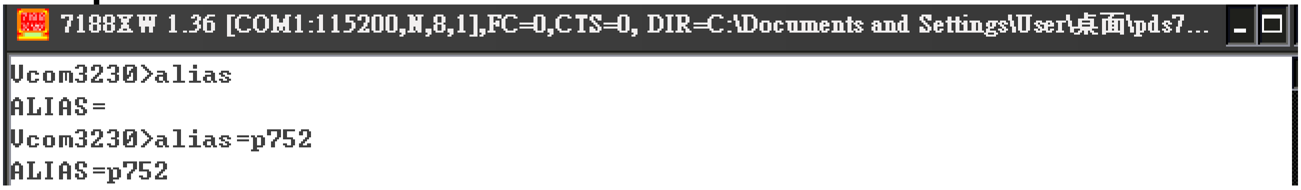
9.4.16 ALIAS

Description: This command is used to query or set the alias of a PDS module.
The maximum character length of PDS alias name is 16 bytes.

Effect: Immediate

Command	Arguments	Description
alias		Queries the alias.
alias	= xxxx	Sets the alias of a PDS module to "xxxx".

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>alias
ALIAS=
Vcom3230>alias=p752
ALIAS=p752
```

9.4.17 DHCP

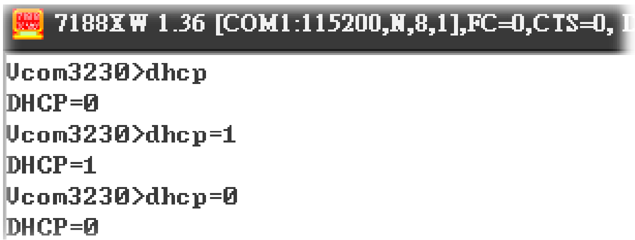
Description: This command is used to set the DHCP client to either enabled or disabled.

DHCP function will get a dynamic IP address setting for PDS automatically. Thus it's recommended to disable DHCP function and use a static IP address setting. This prevents you to configure virtual COM mappings again and again.

Effect: Immediate

Command	Arguments	Description
DHCP	= 0	Disables the DHCP client.
DHCP	= 1	Enables the DHCP client.

Example:



```
Vcom3230>dhcp
DHCP=0
Vcom3230>dhcp=1
DHCP=1
Vcom3230>dhcp=0
DHCP=0
```

9.4.18 UDP

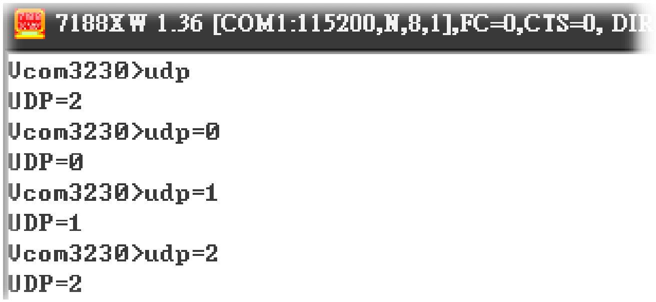
Description: This command is used to configure the UDP Search function.

UDP is used to set the action mode for when a PDS module receives a UDP search command.

Effect: Immediate

Command	Arguments	Description
UDP	= 0	Rejects UDP search commands. The PDS will not reply to the UDP search command, and can not be searched again.
UDP	= 1	Replies to UDP search commands. The PDS modules will reply to the UDP search command, and can be searched.
UDP	= 2 (Default)	Replies to UDP search commands till a client is connected.

Example:



```
Vcom3230>udp
UDP=2
Vcom3230>udp=0
UDP=0
Vcom3230>udp=1
UDP=1
Vcom3230>udp=2
UDP=2
```

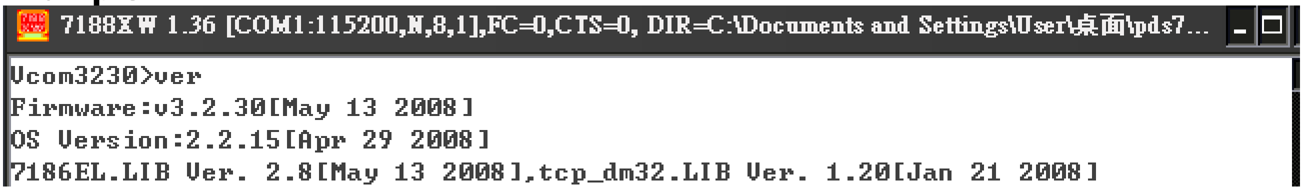
9.4.19 VER

Description: This command is used to query the version information for a PDS module.

Effect: Setting the version information is not allowed.

Command	Argument	Description
VER		Queries the version information.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>ver
Firmware:v3.2.30[May 13 2008]
OS Version:2.2.15[Apr 29 2008]
7186EL.LIB Ver. 2.8[May 13 2008],tcp_dm32.LIB Ver. 1.20[Jan 21 2008]
```

9.4.20 SAVE

Description: This command is used to set the PDS module to backup or no-backup “autoexec.bat” and “vcom.ini” files when using “load” command.

Effect: Immediate

Command	Arguments	Description
save	= 1	When the “load” command is used, backup copies of the "autoexec.bat" and "vcom.ini" files will be saved.
save	= 0 (Default)	When the “load” command is used, backup copies of the "autoexec.bat" and "vcom.ini" files will NOT be saved.

Example: see images [21-1] and [21-2] below.

9.4.21 LOAD

Description: This command is used to loads file to the built-in flash disk on PDS module. It should be used to update firmware only.

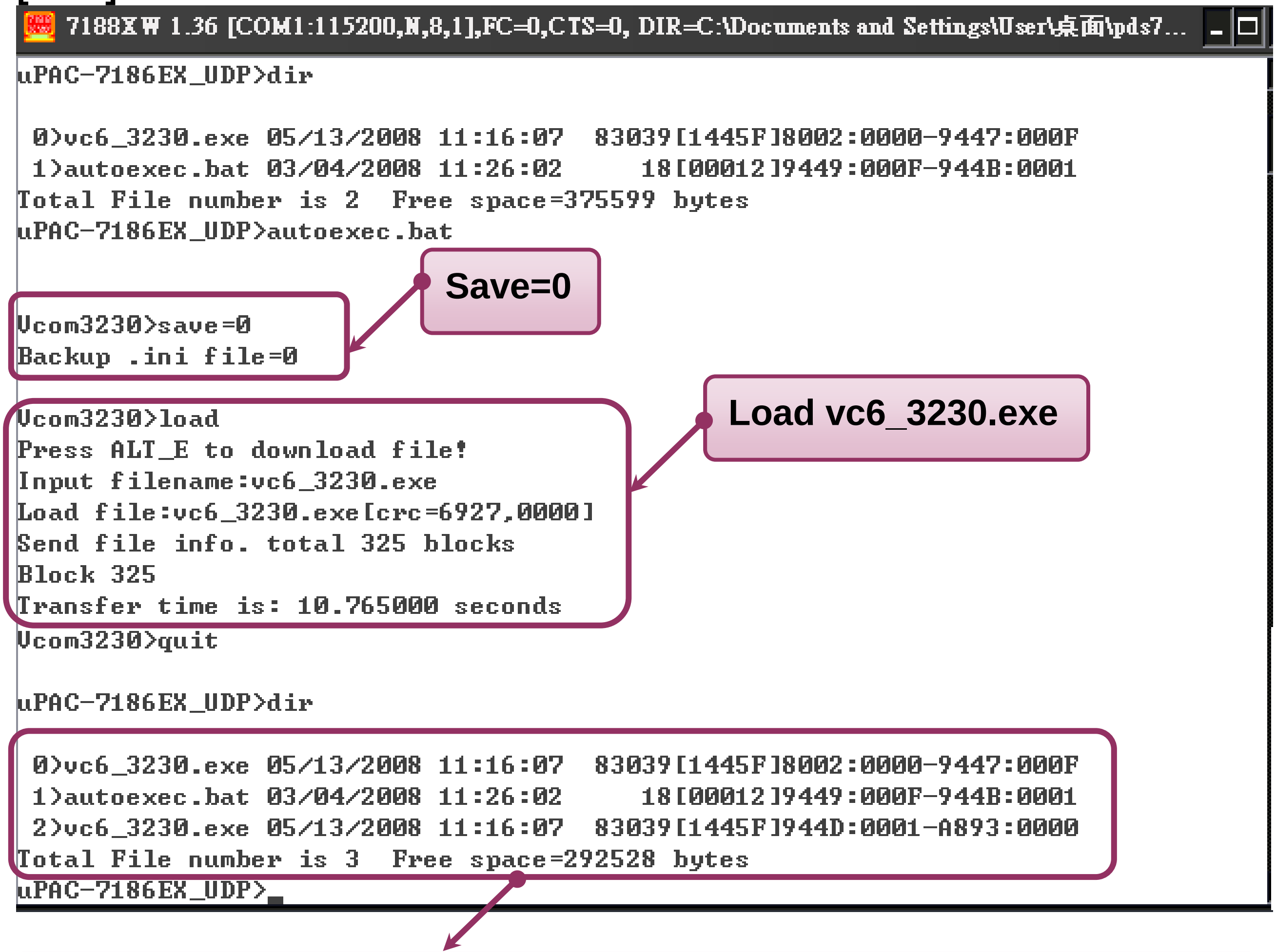
Effect: Immediate

Command	Argument	Description
load		The command is coordinated with the MiniOS7 "load" command and can be used to renew the "vcom3230.exe," "vcom.ini" or "autoexec.bat" file(s).

※"Load" is not a Telnet command.

Example:

[21-1]

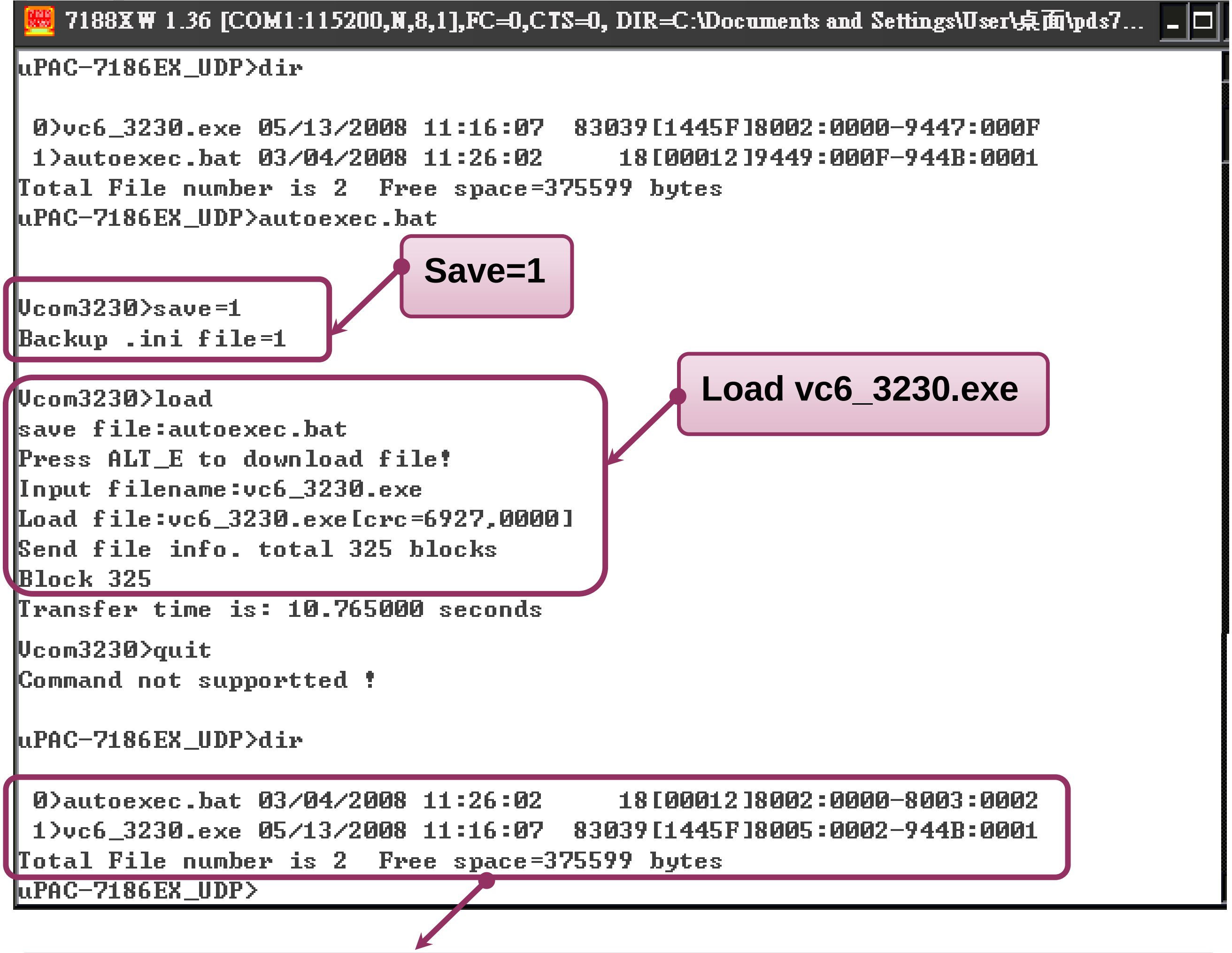


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...
Save=0
Load vc6_3230.exe

```
uPAC-7186EX_UDP>dir

 0)vc6_3230.exe 05/13/2008 11:16:07  83039[1445F]8002:0000-9447:000F
 1)autoexec.bat 03/04/2008 11:26:02     18[00012]9449:000F-944B:0001
Total File number is 2  Free space=375599 bytes
uPAC-7186EX_UDP>autoexec.bat


Vcom3230>save=0
Backup .ini file=0

Vcom3230>load
Press ALT_E to download file!
Input filename:vc6_3230.exe
Load file:vc6_3230.exe[crc=6927,0000]
Send file info. total 325 blocks
Block 325
Transfer time is: 10.765000 seconds
Vcom3230>quit

uPAC-7186EX_UDP>dir

 0)vc6_3230.exe 05/13/2008 11:16:07  83039[1445F]8002:0000-9447:000F
 1)autoexec.bat 03/04/2008 11:26:02     18[00012]9449:000F-944B:0001
 2)vc6_3230.exe 05/13/2008 11:16:07  83039[1445F]944D:0001-A893:0000
Total File number is 3  Free space=292528 bytes
uPAC-7186EX_UDP>_
```

※When save = 0, the system doesn't back up the "autoexec.bat" and "vcom.ini" files to memory, and doesn't clear the flash disk. It only loads the file that is selected specified.

[21-2]


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
uPAC-7186EX_UDP>dir

 0)vc6_3230.exe 05/13/2008 11:16:07  83039[1445F]8002:0000-9447:000F
 1)autoexec.bat 03/04/2008 11:26:02     18[00012]9449:000F-944B:0001
Total File number is 2  Free space=375599 bytes
uPAC-7186EX_UDP>autoexec.bat


Vcom3230>save=1
Backup .ini file=1

Vcom3230>load
save file:autoexec.bat
Press ALT_E to download file!
Input filename:vc6_3230.exe
Load file:vc6_3230.exe[crc=6927,0000]
Send file info. total 325 blocks
Block 325
Transfer time is: 10.765000 seconds

Vcom3230>quit
Command not supportted !

uPAC-7186EX_UDP>dir

 0)autoexec.bat 03/04/2008 11:26:02     18[00012]8002:0000-8003:0002
 1)vc6_3230.exe 05/13/2008 11:16:07  83039[1445F]8005:0002-944B:0001
Total File number is 2  Free space=375599 bytes
uPAC-7186EX_UDP>
```

Save=1

Load vc6_3230.exe

※ When save = 1, system will back up the "autoexec.bat" and "vcom.ini" files to memory first, clear all files in the flash disk, and then load the "autoexec.bat" and "vcom.ini" files from memory and run the "load" command to load the specified file(s).

9.4.22 CONFIG

Description: This command is used to clear the settings in the EEPROM.

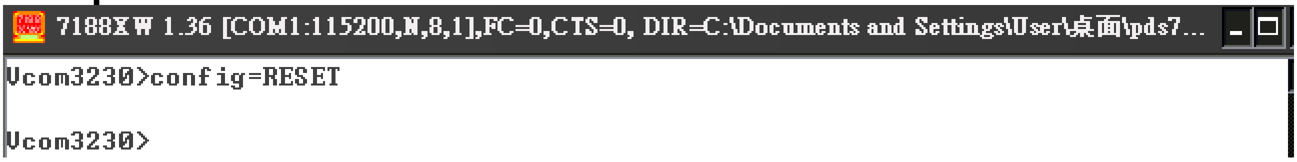
Effect: Immediate

Command	Argument	Description
config	= RESET	Clears the settings in the EEPROM. After reboot, the firmware will use the new (default) settings on EEPROM. Note: ("RESET" MUST be in capital letters.)

※When "Config=RESET" is used, the Password, Alias and IPFILTER settings will also be cleared, but the IP/MASK/GATEWAY addresses will not.

※The SystemTimeout setting is also cleared to 0 by the "config=RESET" command, you have to configure the SystemTimeout value again. The default factory setting of SystemTimeout value should be 300000ms (= 300 seconds).

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>config=RESET

Vcom3230>
```

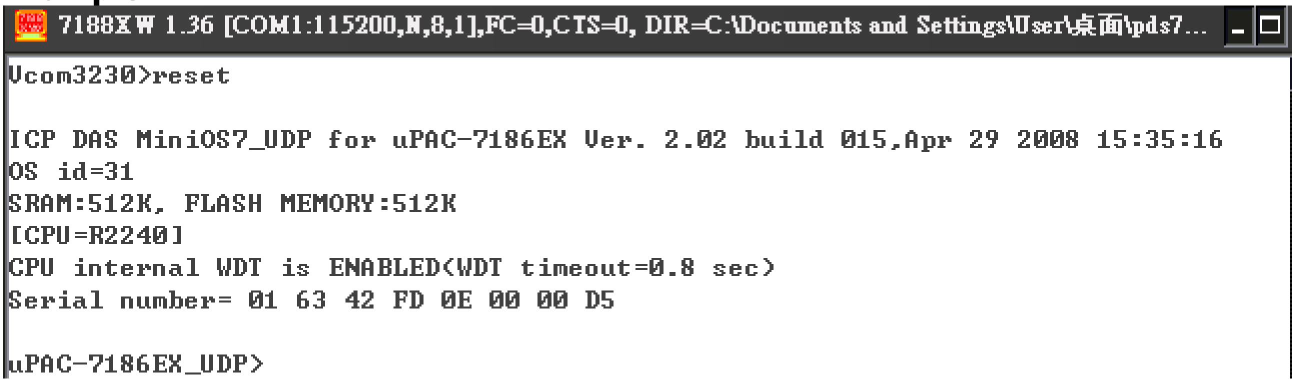
9.4.23 RESET

Description: This command is used to reboot the PDS module.

Effect: Immediate

Command	Argument	Description
reset		Reboot the PDS series module.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>reset

ICP DAS MiniOS7_UDP for uPAC-7186EX Ver. 2.02 build 015,Apr 29 2008 15:35:16
OS id=31
SRAM:512K, FLASH MEMORY:512K
[CPU=R2240]
CPU internal WDT is ENABLED(WDT timeout=0.8 sec)
Serial number= 01 63 42 FD 0E 00 00 D5

uPAC-7186EX_UDP>
```

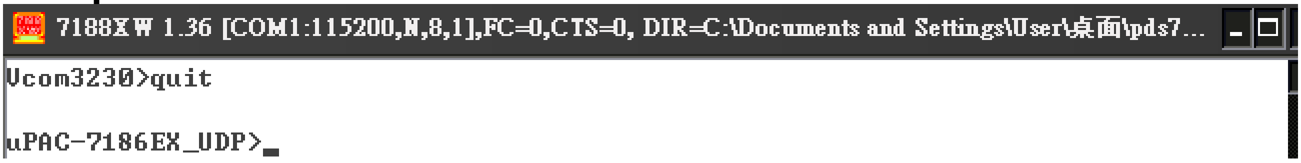
9.4.24 QUIT

Description: This command is used to stop and quit the firmware of PDS series module.

Effect: Immediate

Command	Argument	Description
quit		Quits the firmware.

Example:


7188XW 1.36 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\Documents and Settings\User\桌面\pds7...

```
Vcom3230>quit

uPAC-7186EX_UDP>_
```

10. Modbus Information for PPDS

10.1 What is Modbus TCP/IP?

Modbus is a communication protocol developed by Modicon in 1979. You can also visit http://www.modbus.org to find more valuable information.

The Different versions of Modbus used today include Modbus RTU (based on serial communication interfaces such as RS485 and RS232), Modbus ASCII and Modbus TCP, which is the Modbus RTU protocol embedded into TCP packets.

Modbus TCP is an internet protocol. The protocol embeds a Modbus frame into a TCP frame so that a connection oriented approach is obtained, thereby making it reliable. The master queries the slave and the slave responds with the reply. The protocol is open and, hence, highly scalable.

10.2 Modbus Message Structure

Modbus devices communicate using a master-slave (client-server) technique in which only one device (the master/client) can initiate transactions (called queries). The other devices (slaves/servers) respond by supplying the requested data to the master, or by taking the action requested in the query.

A query from a master will consist of a slave address (or broadcast address), a function code defining the requested action, any required data, and an error checking field. A response from a slave consists of fields confirming the action taken, any data to be returned, and an error checking field.

Modbus/TCP Message Structure

Byte 00~05	Byte 06~11
6-byte header	**RTU Data**

Leading 6 bytes of Modbus/TCP protocol:

Byte 00	Byte 01	Byte 02	Byte 03	Byte 04	Byte 05
Transaction identifier		Protocol identifier		Length field (upper byte)	Length field (lower byte)

Transaction identifier = Assigned by Modbus/TCP master (client)

Protocol identifier = 0

Length field (upper byte) = 0 (since all messages are smaller than 256)

Length field (lower byte) = Number of following RTU data bytes

RTU Data Structure

Byte 06	Byte 07	Byte 08-09	Byte 10-11
Net ID (Station number)	Function Code	Data Field	
		Reference number (Address Mapping)	Number of points

Net ID specifies the address of the receiver (Modbus/TCP slave).

Function Code specifies the message type.

Data Field is the data block.

Net ID (Station Number)

The first byte in the Modbus structure of the Modbus RTU is the receiver's address. The Valid addresses are in the range of 0 to 247. Addresses 0 is used for broadcast, while addresses 1 to 247 are given to individual Modbus devices.

The Net ID should be assigned to the address for the target device you want to read from and write to. For example, if you want to use the built-in I/O of the PPDS, assign the Net ID to the PPDS's ID; if the PPDS is used as a Modbus/TCP to Modbus/RTU gateway, assign the Net ID to the address for the RTU slave device.

Function Code

The second byte in the frame structure of the Modbus RTU is the function code. The function code describes what the slave is required to do. Valid function codes are between 1 and 255. The slave uses the same function code as the request to answer it. Only when an error occurs in the system will the highest bit of the function code be set to '1'. Hence the master will know whether the message has been transmitted correctly or not.

Section	**Code**	**Function**	**Reference (Address)**
10.2.1	**01 (0x01)**	Read Coils status (Readback DOs)	0xxxx
10.2.2	**02 (0x02)**	Read Input Status (Read DIs)	1xxxx
10.2.3	**05 (0x05)**	Force Single Coil (Write DO)	0xxxx
10.2.4	**15 (0x0F)**	Force Multiple Coils (Write DOs)	0xxxx

Data

Data is transmitted in 8-, 16- and 32-bit format. The data of 16-bit registers is transmitted in high-byte first (For example: 0x0A0B ==> 0x0A, 0x0B). The data of 32-bit registers is transmitted as two 16-bit registers, and is low-word first. (For example: 0x0A0B0C0D ==> 0x0C, 0x0D, 0x0A, 0x0B.)

10.2.1 01 (0x01) Read Coils Status (Readback DOs)

This function code is used to read the current coils status or the current digital output readback value.

[Request]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1~247
01	Function code	1 Byte	0x01
02-03	Starting DO address	2 Bytes	Refer to Modbus Address Table for PPDS series. (Sec. 10.3 Modbus Register Map) Byte 02 = high byte Byte 03 = low byte
04-05	Number of points (channels)	2 Bytes	Byte 04 = high byte Byte 05 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x01
02	Byte Count	1 Byte	Byte count of response (n = (Points+7)/8)

03	Data	n Byte	n= 1; Byte 03 = data bit 7~0
			n= 2; Byte 04 = data bit 15~8
			....
			n= m; Byte m+2 = data bit (8m-1)~ 8(m-1)

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x81
02	Exception code	1 Byte	Refer to Modbus Standard Specification for more details

- *For example: Function 01 (0x01), Readback DOs*

[Leading 6 bytes] **[Request]**
Command: 01 02 00 00 00 06 01 01 00 00 00 07

[Leading 6 bytes] **[Response]**
Response: 01 02 00 00 00 04 01 01 01 04

Note: Reads digital output value of PPDS-721D-MTCP

Descriptions as follows:

Command:

[Leading 6 bytes]
Byte 00-03: 01 02 00 00 (Message number)
Byte 04-05: 00 06 (Number of remaining bytes in this frame)

[Request]
Byte 00 : 01 (Net ID)
Byte 01 : 01 (Function Code)
Byte 02-03: 00 00 (Starting DO address)
Byte 04-05: 00 07 (Number of points)

Response:

[Leading 6 bytes]
Byte 00-03: 01 02 00 00 (Message number)
Byte 04-05: 00 04 (Number of remaining bytes in this frame)

[Response]
Byte 00: 01 (Net ID)
Byte 01: 01 (Function Code)
Byte 02: 01 (Byte count of response)
Byte 03: 04 (DO6~DO0 Value)

10.2.2 02 (0x02) Read Input Status (Read DIs)

This function code is used to read the current digital input value.

[Request]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x02
02-03	Starting DI address	2 Bytes	Refer to Modbus Address table for PPDS series. (Sec. 10.3 Modbus Register Map) Byte 02 = high byte Byte 03 = low byte
04-05	Number of points (channels)	2 Bytes	Byte 04 = high byte Byte 05 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x02
02	Byte Count	1 Byte	Byte count of response (n =(Points+7)/8)

03	Data	n Byte	n= 1; Byte 03 = data bit 7~0
			n= 2; Byte 04 = data bit 15~8
			....
			n= m; Byte m+2 = data bit (8m-1)~ 8(m-1)

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x82
02	Exception code	1 Byte	Refer to Modbus Standard Specification for more details

- *For example: Function 02 (0x02), Read DIs*

[Leading 6 bytes] **[Request]**
Command: 01 02 00 00 00 06 01 02 00 00 00 06

[Leading 6 bytes] **[Response]**
Response: 01 02 00 00 00 04 01 02 01 0F

Note: Reads digital input value of PPDS-721D-MTCP

Descriptions as follows:

Command:

[Leading 6 bytes]
Byte 00-03: 01 02 00 00 (Message number)
Byte 04-05: 00 06 (Number of remaining bytes in this frame)

[Request]
Byte 00 : 01 (Net ID)
Byte 01 : 02 (Function Code)
Byte 02-03: 00 00 (Starting DI address)
Byte 04-05: 00 06 (Number of points)

Response:

[Leading 6 bytes]
Byte 00-03: 01 02 00 00 (Message number)
Byte 04-05: 00 04 (Number of remaining bytes in this frame)

[Response]
Byte 00: 01 (Net ID)
Byte 01: 02 (Function Code)
Byte 02: 01 (Byte count of response)
Byte 03: 3B (DI5~DI0 Value)

10.2.3 05 (0x05) Force Single Coil (Write DO)

This function code is used to set a single coil status or a single digital output value.

[Request]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x05
02-03	DO Address	2 Bytes	Refer to Modbus Address Table for PPDS series. (Sec. 10.3 Modbus Register Map) Byte 02 = high byte Byte 03 = low byte
04-05	Output value	2 Bytes	0xFF 00 set the output to ON. 0x00 00 set the output to OFF. All other values are illegal and will not affect the coil. Byte 04 = high byte Byte 05 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x05
02-03	DO Address	2 Bytes	The value is the same as Byte 02-03 of the Request
04-05	Output value	2 Bytes	The value is the same as Byte 04-05 of the Request

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x85
02	Exception code	1 Byte	Refer to Modbus Standard Specification for more details

- *For example: function 05 (0x05), Write DO*

[Leading 6 bytes] **[Request]**
Command: 01 02 00 00 00 06 01 05 00 02 FF 00

[Leading 6 bytes] **[Response]**
Response: 01 02 00 00 00 06 01 05 00 02 FF 00

Note: Sets the DO2 to ON of PPDS-721D-MTCP

Descriptions as follows:

Command:

[Leading 6 bytes]
Byte 00-03: 01 02 00 00 (Message number)
Byte 04-05: 00 06 (Number of remaining bytes in this frame)

[Request]
Byte 00 : 01 (Net ID)
Byte 01 : 05 (Function Code)
Byte 02-03: 00 02 (DO address)
Byte 04-05: FF 00 (Set the output to ON)

Response:

[Leading 6 bytes]
Byte 00-03: 01 02 00 00 (Message number)
Byte 04-05: 00 06 (Number of remaining bytes in this frame)

[Response]
Byte 00 : 01 (Net ID)
Byte 01 : 05 (Function Code)
Byte 02-03: 00 02 (DO address)
Byte 04-05: FF 00 (Set the output to ON)

10.2.4 15 (0x0F) Force Multiple Coil (Write DOs)

This function code is used to set multiple coils status or write multiple digital output value.

[Request]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x0F
02-03	Starting DO address	2 Bytes	Refer to Modbus Address Table for PPDS series. (Sec. 10.3 Modbus Register Map) Byte 02 = high byte Byte 03 = low byte
04-05	Number of output channels (Points)	2 Bytes	Byte 04 = high byte Byte 05 = low byte
06	Byte count	1 Byte	n = (Points +7)/8

07	Output value	n Byte	A bit corresponds to a channel. Value 1 for a bit denotes the channel is ON, while the value 0 is OFF.
			n= 1; Byte 07 = data bit 7 ~ 0
			n= 2; Byte 08 = data bit 15 ~ 8
			....
			n= m; Byte m+6 = data bit (8m-1)~ 8(m-1)

[Response]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x0F
02-03	Starting DO address	2 Bytes	The value is the same as Byte 02-03 of the Request
04-05	Number of output channels (Points)	2 Bytes	The value is the same as Byte 04-05 of the Request

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station number)	1 Byte	1-247
01	Function code	1 Byte	0x8F
02	Exception code	1 Byte	Refer to Modbus Standard Specification for more details

- *For example: function 0F (0x0F), Write DOs*

```
          [Leading 6 bytes]    [Request]
Command:  01 02 00 00 00 08   01 0F 00 00 00 07 01 7F

          [Leading 6 bytes]    [Response]
Response: 01 02 00 00 00 06   01 0F 00 00 00 07
```

Note: Sets DO0-DO6 to NO of PPDS-721D-MTCP

Descriptions as follows:

Command:

[Leading 6 bytes]
Byte 00-03: 01 02 00 00 (Message number)
Byte 04-05: 00 08 (Number of remaining bytes in this frame)

[Request]
Byte 00 : 01 (Net ID)
Byte 01 : 0F (Function Code)
Byte 02-03: 00 00 (Starting DO address)
Byte 04-05: 00 07 (Number of output channels)
Byte 06 : 01 (Byte count)
Byte 07 : 7F (Output value)

Response:

[Leading 6 bytes]
Byte 00-03: 01 02 00 00 (Message number)
Byte 04-05: 00 06 (Number of remaining bytes in this frame)

[Response]
Byte 00 : 01 (Net ID)
Byte 01 : 0F (Function Code)
Byte 02-03: 00 00 (Starting DO address)
Byte 04-05: 00 07 (Number of output channels)

Data Field

The data field of messages sent between a master and a slave contains additional information about the action to be taken by the master or any information requested by the slave. If the master does not require this information, the data field can be empty.

Reference (Address)	Description
0xxxx	Read/Write Discrete Outputs or Coils. A 0x reference address is used to output device data to a digital output channel.
1xxxx	Read Discrete Inputs. The ON/OFF status of a 1x reference address is controlled by the corresponding digital input channel.

For more detail of Address Mapping (Reference number), refer to 10.3 Modbus Register Map.

10.3 Modbus Register Map

The nDI and nDO parameters of each model used in the following Modbus address tables are shown as follows:

Model name	Number of DO channels **(nDO)**	Number of DI channels **(nDI)**
PPDS(M)-721(D)-MTCP	7	6
PPDS(M)-732(D)-MTCP	4	4
PPDS(M)-734(D)-MTCP	4	4
PPDS(M)-743(D)-MTCP	4	4
PPDS(M)-762(D)-MTCP	2	1

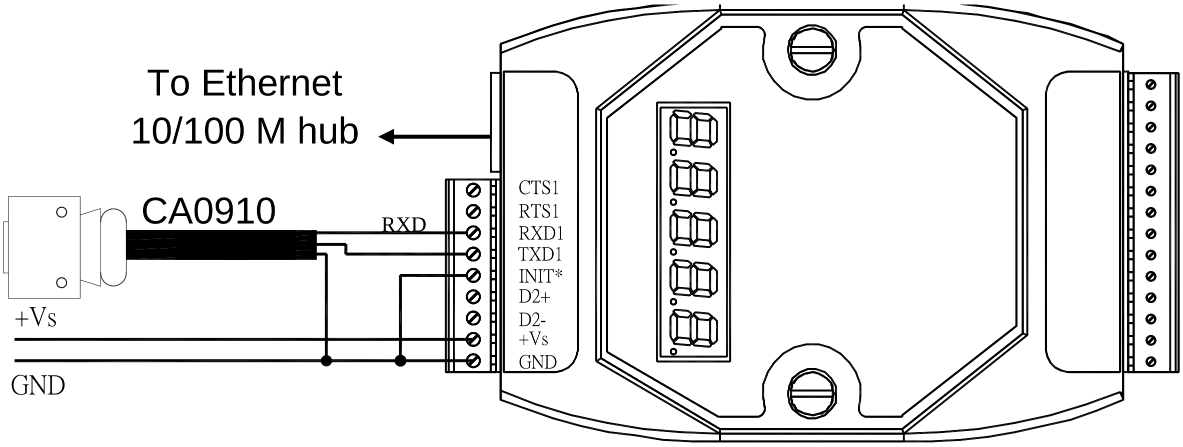
- **0xxxx: DO address (base 0)**

Begin address		Points	Description	Range	Access Type	Factory Set Value
DEC	0~(nDO-1)	1~nDO	Digital Output Value	0 = Off 1 = On	R/W	-
HEX	0x00~0x(nDO-1)					
"**R**": Read; "**W**": Write						

- **1xxxx: DI address (base 1)**

Begin address		Points	Description	Range	Access Type
DEC	0 ~ (nDI-1)	1~nDI	Digital Input Value	0 = Off 1 = On	R
HEX	0x00 ~ 0x(nDI-1)				
"**R**": Read					

Appendix: Linking to a Development PC



Step 1: Connect the download-cable, CA0910, between the PDS series module and COM 1 (or COM 2) of the development PC as per above the diagram.

Step 2: Connect the INIT* pin to the GND pin, as shown in the above diagram.

Step 3: Unzip the "**7188XW_yyyymmdd.zip**" file on the PC. The file is located in the CD:\Napdos\MiniOS7\utility folder.

Step 4: Apply power (+Vs, GND) to the PDS series module. The +Vs can be anywhere from +30 ~ +10 V. The PPDS(M)-700-MTCP, PPDS-700-IP67, DS-700, PDS-782-25 and PDS-5105D-MTCP module +Vs can be anywhere from +48 ~ +12 V.

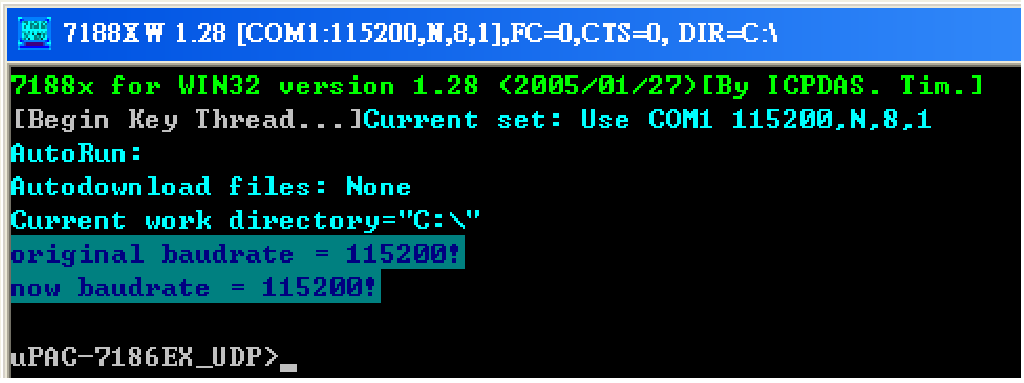
Step 5: Check that the 5-digit 7-SEG LED is continuously showing the following information:

Hours.Minutes.Seconds

! **Note:** Only display versions of PDS series modules have a 5-digit 7-SEG LED.

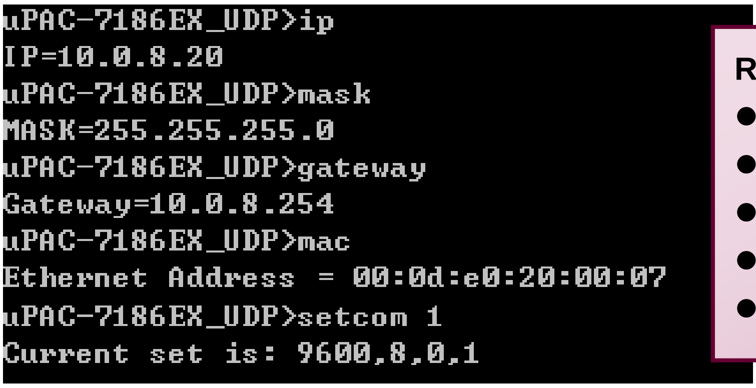
Step 6: Execute 7188XW.EXE/C#, and change the Baud Rate to 115200 bps, N81. "/C#" is the COM Port of the development PC.

Step 7: Press **[Enter]** twice on the development PC:



```
7188XW 1.28 [COM1:115200,N,8,1],FC=0,CTS=0, DIR=C:\
7188x for WIN32 version 1.28 (2005/01/27)[By ICPDAS. Tim.]
[Begin Key Thread...]Current set: Use COM1 115200,N,8,1
AutoRun:
Autodownload files: None
Current work directory="C:\"
original baudrate = 115200!
now baudrate = 115200!

uPAC-7186EX_UDP>_
```

Step 8: Read the configuration of the PDS:

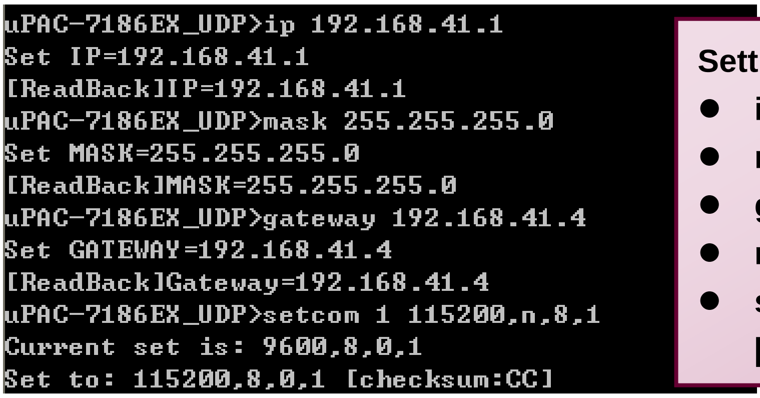


```
uPAC-7186EX_UDP>ip
IP=10.0.8.20
uPAC-7186EX_UDP>mask
MASK=255.255.255.0
uPAC-7186EX_UDP>gateway
Gateway=10.0.8.254
uPAC-7186EX_UDP>mac
Ethernet Address = 00:0d:e0:20:00:07
uPAC-7186EX_UDP>setcom 1
Current set is: 9600,8,0,1
```

Read configuration command

- ip
- mask
- gateway
- mac
- setcom port

Note: The configuration of the PDS as follows can be changed:



```
uPAC-7186EX_UDP>ip 192.168.41.1
Set IP=192.168.41.1
[ReadBack]IP=192.168.41.1
uPAC-7186EX_UDP>mask 255.255.255.0
Set MASK=255.255.255.0
[ReadBack]MASK=255.255.255.0
uPAC-7186EX_UDP>gateway 192.168.41.4
Set GATEWAY=192.168.41.4
[ReadBack]Gateway=192.168.41.4
uPAC-7186EX_UDP>setcom 1 115200,n,8,1
Current set is: 9600,8,0,1
Set to: 115200,8,0,1 [checksum:CC]
```

Settings configuration command

- ip [new ip]
- mask [new mask]
- gateway [new gateway]
- mac [new mac]
- setcom port [baud][data_bit][parity][stop_bit]

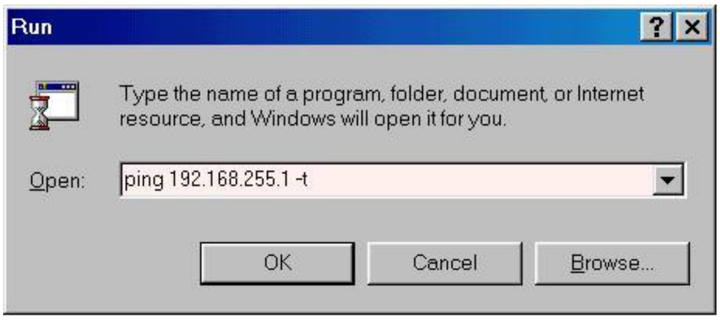
"setcom" parameters are as follows:

Port	1 - 8
Baud	2 - 921600
DataBit	7, 8: for COM 1 and COM 2 5,6,7,8: for COM 3 ~ COM 8
Parity	N, n : None parity E, e : Even parity O, o : Odd parity M, m : Mark, parity = 1 S, s : Space, parity = 0
StopBit	1: for COM 1, COM 2 1, 2: for COM 3 ~ COM 8

Step 9: Disconnect the INIT* pin from the GND pin.

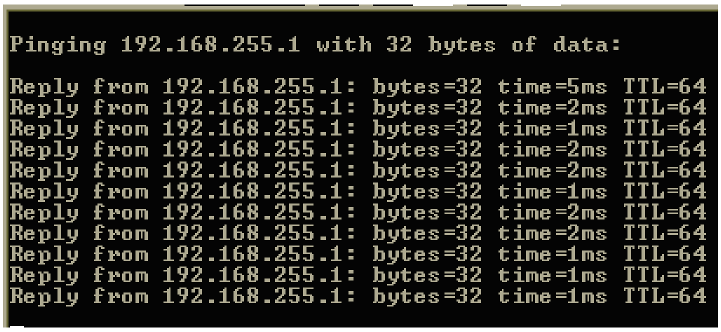
Step 10: Power-off the module then power on again.

Step 11: Execute **ping 192.168.255.1 –t** using a run command as follows:


Run
? ×
Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.
Open:
OK
Cancel
Browse...

Open: ping 192.168.255.1 -t

Execute Ping 192.168.255.1 on the PC client.



```
Pinging 192.168.255.1 with 32 bytes of data:

Reply from 192.168.255.1: bytes=32 time=5ms TTL=64
Reply from 192.168.255.1: bytes=32 time=2ms TTL=64
Reply from 192.168.255.1: bytes=32 time=1ms TTL=64
Reply from 192.168.255.1: bytes=32 time=2ms TTL=64
Reply from 192.168.255.1: bytes=32 time=2ms TTL=64
Reply from 192.168.255.1: bytes=32 time=1ms TTL=64
Reply from 192.168.255.1: bytes=32 time=2ms TTL=64
Reply from 192.168.255.1: bytes=32 time=2ms TTL=64
Reply from 192.168.255.1: bytes=32 time=1ms TTL=64
Reply from 192.168.255.1: bytes=32 time=1ms TTL=64
Reply from 192.168.255.1: bytes=32 time=1ms TTL=64
```

The ping results should be smooth and continuous

Note:

- **192.168.255.1** is the default IP of the PDS series module. The IP address can be changed using the instructions in step 8.
- If the PDS cannot be successfully pinged from the PC, refer to step 8 to change the configuration of the PDS series module. (The mask and gateway addresses of PDS series module and the PC should make the network definition.)
- The MAC address of the PDS series module should be unique on the same network. Refer to step 8 for details of how to change the MAC address of the PDS series module.
- Each PDS MAC address is unique in the default shipping.

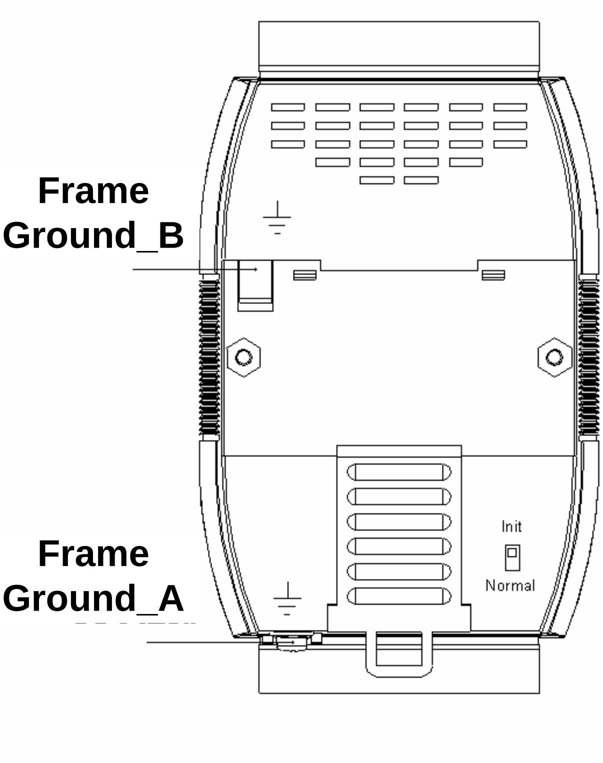
In general, if the host PC can ping the PDS series module smoothly and continuously, all other software and drivers for the PDS series module will operate correctly. Therefore, users should ensure that the development PC is able to ping the PDS series module smoothly before any further testing is carried out.

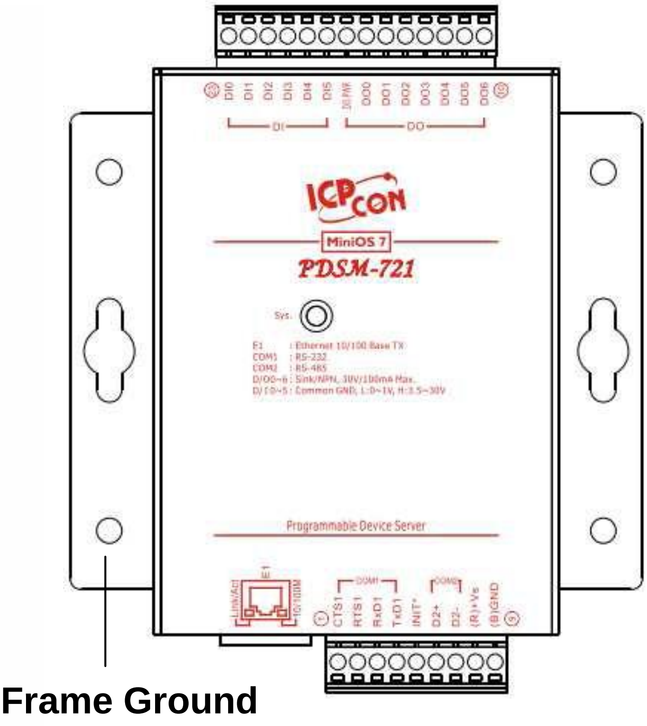
Appendix: Frame Ground

Electronic circuits are constantly vulnerable to Electro Static Discharge (ESD), which becomes worse in a continental climate area. PDS series modules feature a new design for the frame ground, which provides a path for bypassing ESD, allowing enhanced static protection (ESD) capability and ensures that the module is more reliable.

It is recommended that the Frame Ground of the PDS series module is corrected to the earth ground, such as the ground of an AC power supply, to provide better ESD protection for the module.

The PDS-700, PPDS-700-MTCP, DS-700 module is designed with two Frame Ground contact points, Frame-Ground-A and Frame-Ground-B, as shown in the figure below. When mounted to a DIN rail, Frame-Ground-B and the DIN rail are in contact. Thus, protection can be achieved by also connecting the DIN rail to earth ground.





Appendix: Glossary

1. ARP (Address Resolution Protocol)

Consider two machines A and B that share a physical network. Each has an assigned IP address IPA and IPB, and a MAC address the MACA and MACB. The goal is to devise low-level software that hides MAC addresses and allows higher-level programs to work only with the IP addresses. Ultimately, however, communication must be carried out by the physical networks using whatever MAC address scheme the hardware supplies.

Suppose machine A wants to send a packet to machine B across a physical network to which they are both attached, but A only has the Internet address for B, IPB. The question arises: how does A map that address to the MAC address for B, MACB?

ARP provides a method of dynamically mapping 32-bit IP address to the corresponding 48-bit MAC address. The term dynamic is used since it happens automatically and is normally not a concern for either the application user or the system administrator.

2. Clients and Servers

The client-server paradigm uses the direction of initiation to categorize whether a program is a client or server. In general, an application program that initiates peer to peer communication is called a client. End users usually invoke client programs when they use network services.

Most client programs consist of conventional application program develop tools. Each time a client program is executed, it contacts a server, sends a request and waits for a response. When the response arrives, the client program continues processing. Client programs are often easier to develop than servers, and usually require no special system privileges to operate.

By comparison, a server is any program that waits for incoming requests from a client program. The server receives a request from a client, performs the necessary computation and returns the result to the client.

3. Ethernet

The term Ethernet generally refers to a standard published in 1982 by Digital Equipment Corp., Intel Corp. and Xerox Corp. Ethernet is the most popular physical layer local area network (LAN) technology today. Ethernet is a best-effort delivery system that uses CSMA/CD technology. It recognizes hosts using 48-bit MAC address.

4. Firmware

Firmware is an alterable program located or stored in the semi-permanent storage area, e.g., ROM, EEPROM, or Flash memory.

5. Gateway

Computers that interconnect two networks and pass packets from one to the other are called Internet Gateways or Internet Routers. Gateways route packets that are based on the destination network, not on the destination host.

6. ICMP (Internet Control Messages Protocol)

No system works correctly all the time. ICMP provides a method of communicating between the Internet Protocol software on one machine and the Internet Protocol software on another. It allows gateways to send error or control messages to other gateways or allows a host to know what is wrong with the network communication.

7. Internet

Physically, the Internet is a collection of packet switching networks interconnected by gateways along with TCP/IP protocol that allows them to perform logically as a single, large and virtual network. The Internet recognizes hosts using 32-bit IP address.

8. IP (Internet Protocol) address

Every interface on an Internet must have a unique IP address (also called an Internet address). These addresses are 32-bit numbers. They are normally written as four decimal numbers, one for each byte of the address such as “192.168.41.1”. This is called dotted-decimal notation.

9. MAC (Media Access Control) address

To allow a computer to determine which packets are meant for it, each computer attached to an Ethernet is assigned a 48-bit integer known as its MAC address (also called an Ethernet address, hardware address or physical address). They are normally written as eight hexadecimal numbers such as “00:71:88:af:12:3e:0f:01”. Ethernet hardware manufacturers purchase blocks of MAC addresses and assign them in sequence as they manufacture the Ethernet interface hardware. Thus, no two hardware interfaces have the same MAC address.

10. Packet

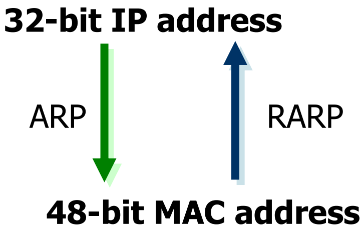
A packet is the unit of data sent across a physical network. It consists of a series of bits containing data and control information, including the source and the destination node (host) address, and is formatted for transmission from one node to another.

11. Ping

Ping sends an ICMP echo request message to a host, expecting an ICMP echo reply to be returned. Normally, if a host cannot be pinged, you won’t be able to use Telnet or FTP to connect to the host. Conversely, if Telnet or FTP cannot be used to connect to a host, Ping is often the starting point to determine what the problem is.

12. RARP (Reverse Address Resolution Protocol)

RARP provides a method of dynamically mapping 48-bit MAC address to the corresponding 32-bit IP address.



13. Socket

Each TCP segment contains the source and destination port number that can be used to identify the sending and receiving application. These two values, along with the source and destination IP address in the IP header, uniquely identify each connection.

The combination of an IP address and a port number is called a socket.

14. Subnet Mask

Subnet mask is often simply called the mask. Given its own IP address and its subnet mask, a host can determine if a TCP/IP packet is destined for a host that is (1) on its own subnet, or (2) on a different network. If (1), the packet will be delivered directly; otherwise if, will be delivered via gateways or routers.

15. TCP (Transmission Control Protocol)

TCP provides a reliable flow of data between two hosts. It is associated with tasks such as dividing the data passed to it from applications into appropriately sized chunks for the network layer below, acknowledging received packets, setting timeouts to make certain that the other end acknowledges packets that are sent, and so on.

16. TCP/IP

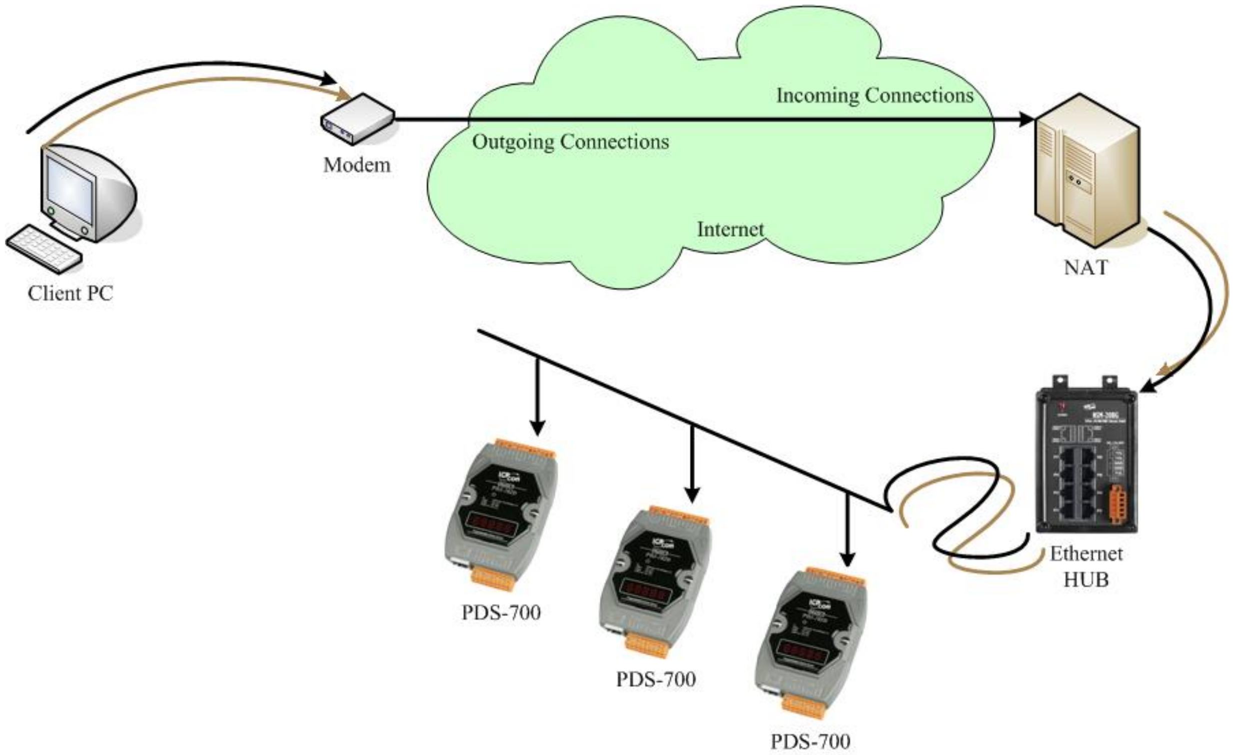
The transmission Control Protocol (TCP) and the Internet Protocol (IP) are the standard network protocols. They are almost always implemented and used together and called TCP/IP. TCP/IP can be used to communicate across any set of interconnected networks.

17. UDP (User Datagram Protocol)

UDP provides a much simpler service to the application layer. It just sends packets of data from one host to the other. But there is no guarantee that the packets will reach the destination host.

Appendix: FAQ

1. How to access the remote PDS that placed behind an NAT or firewall?



The remote site must have a NAT (or a router supports NAT) server. NAT stands for Network Address Translator.

By using (configuring) the NAT server, NAT can forword (bypass) all specified TCP port connection to specified PDS devices.

For example:

NAT: 10000 ~ 10008 maps to 192.168.1.101: 10000 ~ 10008
NAT: 10010 ~ 10018 maps to 192.168.1.102: 10000 ~ 10008

Please note, if your NAT (router) built-in a firewall feature, you have to configure the NAT to allow incoming TCP port connections.

For example:

TCP port includes 10000 ~ 10008 and 10010 ~ 10018 of NAT.

In the VxComm Utility, you have to add PDS by using NAT's address and NAT's TCP ports instead of PDS's setting.

For example:

To add first PDS, it's IP: Port should be NAT: 10000.
To add second PDS, it's IP: Port should be NAT: 10010.

2. How to open a virtual COM port that larger than "COM 9" by calling CreateFile() Win32 API?

If you want to open "COM 10", the correct way to call the CreateFile() is as follows:

```
CreateFile(
                "\\\\. \\COM10",        // address of name of the communications device
                fdwAccess,              // access (read-write) mode
                0,                      // share mode
                NULL,                   // address of security descriptor
                OPEN_EXISTING, // how to create
                0,                      // file attributes
                NULL                    // handle of file with attributes to copy
                );
```

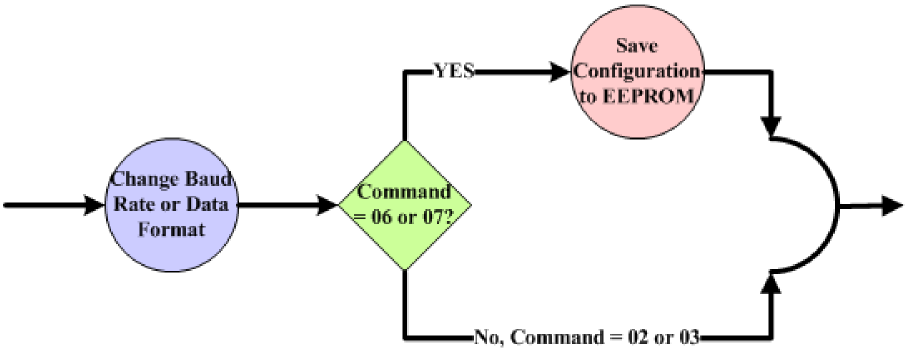
NOTES:

1. This syntax also works for ports COM 1 through COM 9. See more... MS Q115831.
2. Maximum COM port number for VxComm Driver is COM 256.
3. Valid COM port number for MSCOMM.OCX is between 1 to 16. Please refer to MSComm.CommPort.
4. The "\\.\" prefix must be add to the COM port name (device name) when it is larger than "COM 9". But please note that the "\" character is a special escape symbol in C\C++ language, thus you have to use "\\\\.\\" prefix in C\C++ language.

3. Does VxComm Driver (PC) v2.00 work with VxComm Server v2.6.00?

No, please upgrade to version 2.6.14 or the latest version of the VxComm Server. Version 2.6.00 of the VxComm Server uses the "**06**" and "**07**" command to change the BaudRate and the data format and then saves the configuration in the EEPROM.

The newer versions include the "**02**" and "**03**" command that is used to change the BaudRate and data format without needing to be saved. These two commands improve the performance of the Server when changing the configuration settings.



The VxComm Driver (PC) has also been changed to enable the new commands to be used. Thus, users must upgrade their VxComm Server to the latest version if the current used firmware is old version (before v2.6.00).

4. Does VxComm Driver (PC) support auto-reconnection after fixing a network break?

Yes, the VxComm Driver (PC) supports the auto-reconnection mechanism in version 2.00 and above. The VxComm Utility allows the user to set the Keep-Alive Time (ms) and Connection-Broken Time (ms) in the server options.

For more details, refer to the "**Adding a 7188E/8000E/PDS-700/DS-700 server and configuring the VxComm Driver**" section of the VxComm Driver/Utility User Manual.

5. Why doesn't the VxComm Driver (PC) receive data from the PDS series module?

Make sure that the PDS series module is operating in mode 0 (/M0). PDS series module has the following two communication modes:

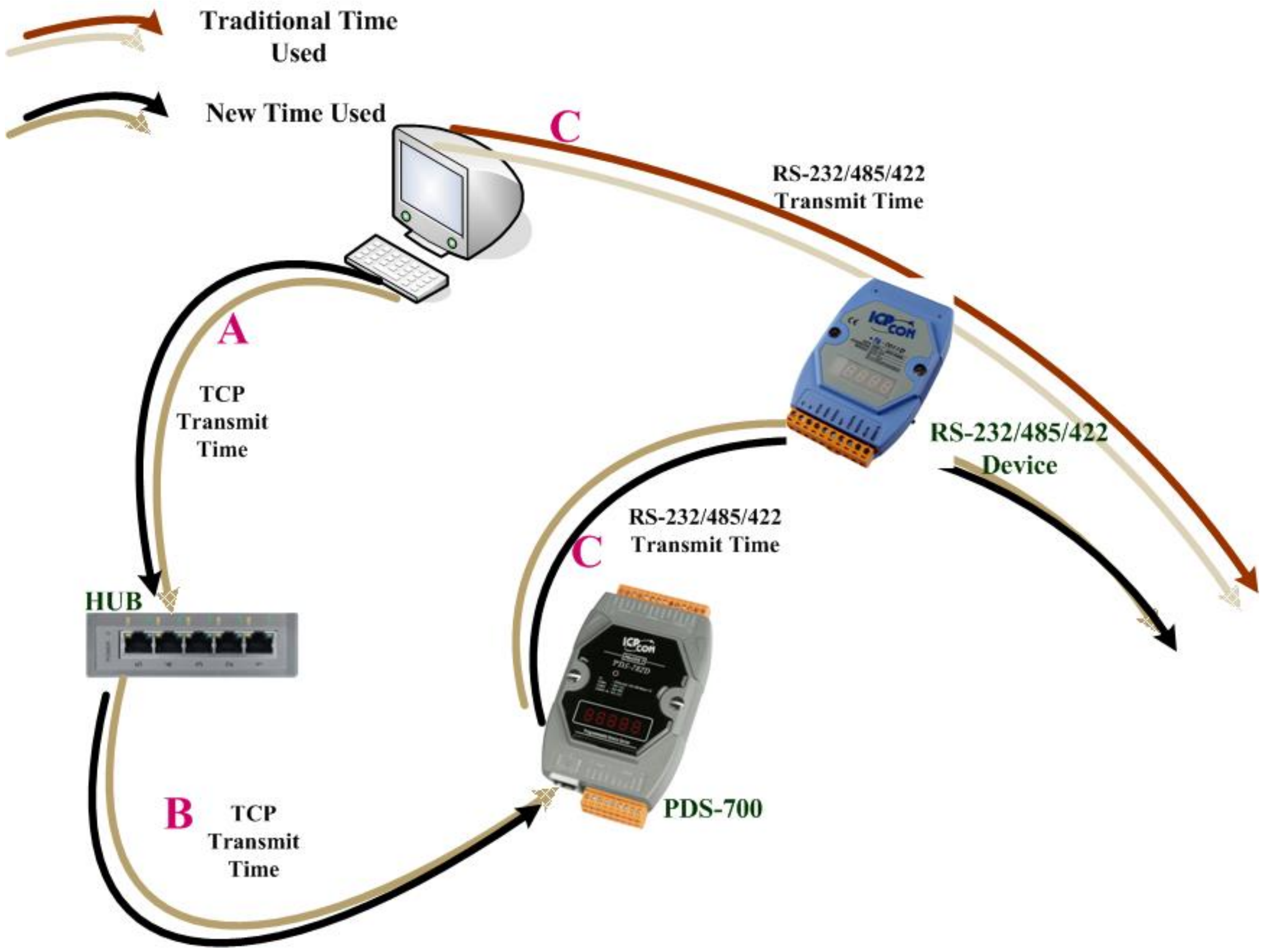
/M0	**Transparent Mode (Multi-echo, shared).** In this mode, data is echoed from the COM Ports of the PDS series module to each client that is connected to the PDS series module.	
/M1	**Slave Mode (Single-echo, Non-Shared).** In this mode, data is echoed from the COM Ports of the PDS series module to the specific client that requested the service.	Version 2.6.12 and above

In /M1 mode, if the client does not send a request to the COM port of the PDS series module, then the module won't return any data to it. For more information, please refer 5.4 ”COM Port Settings” section.

Other reasons causing the problem may be: incorrect wiring, power supply problems IP conflicts, MAC conflicts, an incorrect subnet mask or an invalid IP address. For more details, refer to the "Diagnostics and Troubleshooting" section of the VxComm Driver/Utility User Manual.

6. Does the transmission speed become faster when the serial device working with Serial to Ethernet device servers?

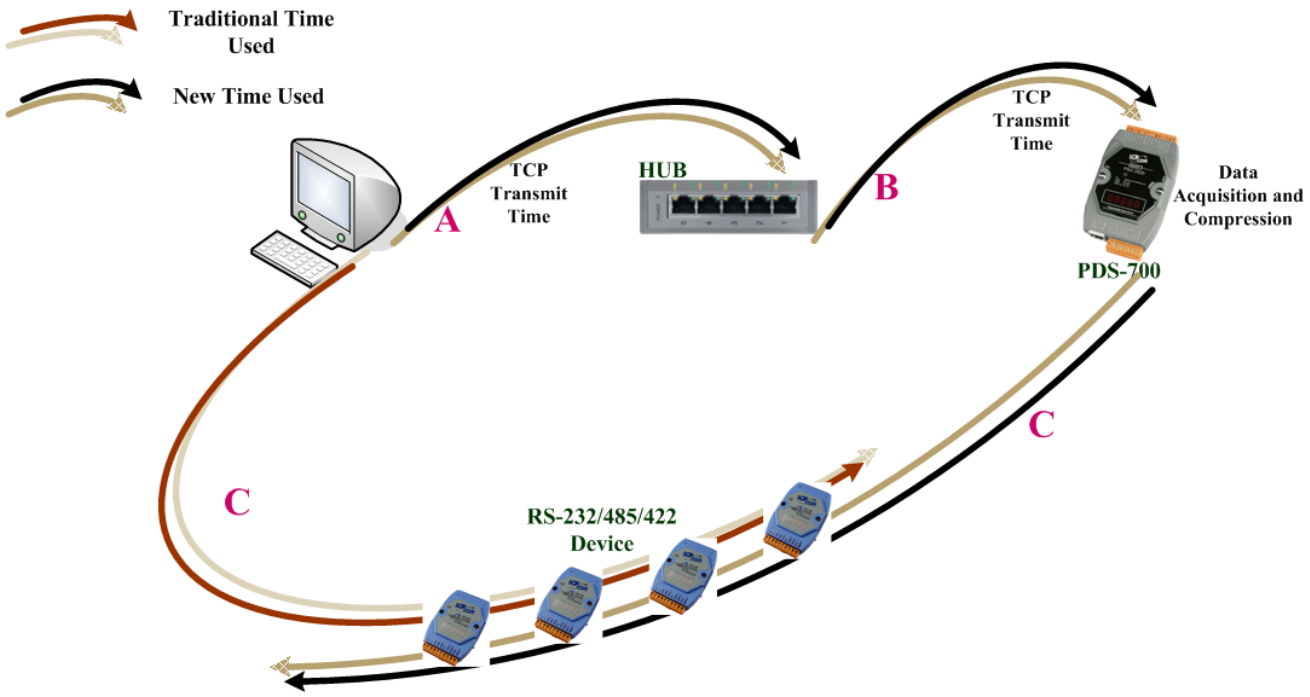
The speed depends on the applications. For transparent applications, it includes Ethernet latency in transmission and may get slower. But you can improve the communication speed by increasing the baud rate since you placing the device server more close to serial device and reduce the communication distance. The higher baud rate should be able to be used in short cable (distance) without communication problem.



Traditional time used	RS-232/485/422 transmit time (C)
New time used	Internet/Ethernet transmit time + RS-232/485/422 transmit time (A+B+C)

(All TCP packets need an extra ACK packet to commit the transmit action. This also causes a little additional delay in communication).

For Xserver applications, it can become faster. User can write their own Xserver applications to acquire data automatically, and then compress and transmit this large amount data at one time. Your application can reach high performance by pre-acquire data before asking by client and then response immediately.



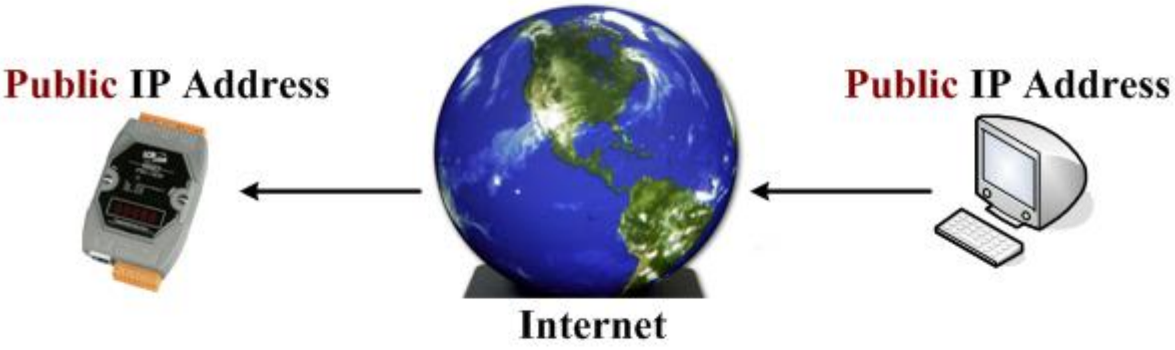
Traditional time used	RS-232/485/422 transmit time (C * n modules)
New time used	Internet/Ethernet transmit time (A + B + C)

7. Why does the PDS series module fail on a (public) Internet connection?

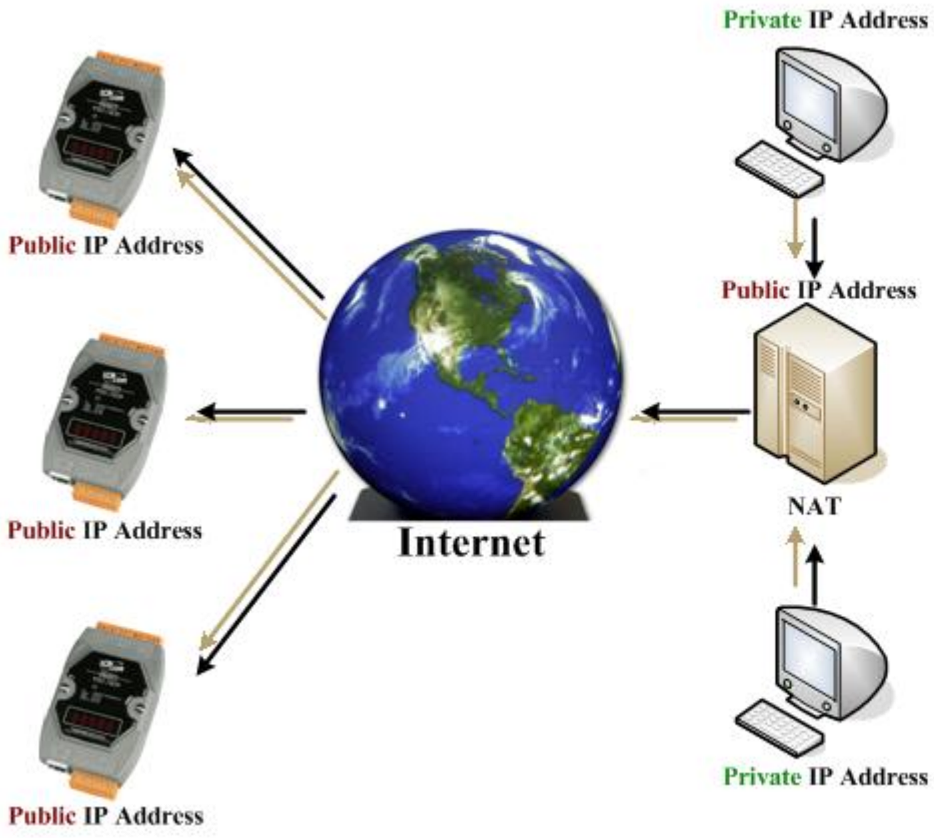
The default IP address of the PDS series module is 192.168.255.1, which can be only used on a private Internet connection. A private network packet will not be routed via a (public) Internet connection, which is the reason why the PDS series module failed on the Internet.

The IANA has reserved three address spaces for private internets (RFC1918).
10.0.0.0 - 10.255.255.255 (10/8 prefix)
172.16.0.0 - 172.31.255.255 (172.16/12 prefix)
192.168.0.0 - 192.168.255.255 (192.168/16 prefix)

The PDS sereis module can operate on the Internet using a legal public IP address. This address can be obtained from your ISP or network administrator.



A private internet client may communicate with a public Internet server (PDS series modules) only if the NAT service for the client is available.



Note:

IANA	Internet Assigned Numbers Authority
RFC	Request for Comments
ISP	Internet Service Providers
NAT	Network Address Translator

8. Can I use the SetCommState ( ) API to changes the Baud Rate/data format settings of a virtual COM port?

Yes. In a Win32 environment, the CreateFile( ) API should be called to open the COM Port(s) and then the SetCommState( ) API can be used to configure the settings.

Third-party tools may provide an OpenCom( ) function for accessing a COM port. In actuality, the CreateFile( ) and SetCommState( ) APIs must be used to implement these kinds of functions.

9. How many PCs can be connected to a single PDS device?

This depends on how many serial ports are available on the PDS series module and how many serial ports which can be connected to each PC of PDS.

The PDS series module has 32 sockets in total includes some reserved listening sockets. The PDS series module provides a single command port for configuring the data (serial) ports. Thus, no matter how many data (serial) ports on the PDS are used, one more socket connection is needed for the command port in order to configure them.

	IO Port	Data Ports	Listening Sockets	Available Sockets	Max. PCs when using all data ports	Max. PCs when using 1 data port
DS-712	0	1	5	32 - 5 = 27	30/2 =15	30/2 = 15
DS-715	0	1	5	32 - 5 = 27	30/2 = 15	30/2 = 15
PDS-720 **PDS-720D**	0	2	6	32 - 6 = 26	29/3 = 9	29/2 = 14
PDS-721 **PDS-721D**	1	2	7	32 - 7 = 25	29/3 = 9	29/2 = 14
PDS-732 **PDS-732D**	1	3	8	32 - 8 = 24	28/4 = 7	28/2 = 14
PDS-734 **PDS-734D**	1	3	8	32 - 8 = 24	28/4 = 7	28/2 = 14
PDS-742 **PDS-742D**	0	4	8	32 - 8 = 24	27/5 = 5	27/2 = 13
PDS-743 **PDS-743D**	1	4	9	32 - 9 = 23	27/5 = 5	27/2 = 13
PDS-752 **PDS-752D**	0	5	9	32 - 9 = 23	26/6 = 4	26/2 = 13

PDS-755 **PDS-755D**	0	5	9	32 - 9 = 23	26/6 = 4	26/2 = 13
PDS-762 **PDS-762D**	1	5	10	32 - 10 = 22	26/6 = 4	26/2 = 13
PDS-782 **PDS-782D**	0	8	12	32 - 12 = 20	23/9 = 2	23/2 = 11
PDS-782-25 **PDS-782D-25**	0	8	12	32 - 13 = 20	23/9 = 2	23/2 = 11

Notes:

1. CMD Port = Command Port (TCP port 10000). The CMD Port is used to configure the data ports (TCP port 10001 ~ 10008) of a PDS series module, such as BaudRate, and data format, etc.
2. The data port (TCP port 10001 ~ 10008, which are mapped to serial ports 1 ~ 8 of PDS), is only used to send/receive data.
3. The Listening Sockets (for PDS series modules) = Number of Data ports + 1 CMD port + IO port + Web + Telnet + UDP Search.
4. The number of Available Sockets (for PDS series modules) = max. (32) sockets - Listening sockets.
5. The maximum number of PCs when using all data ports of PDS = Available sockets/(data ports + 1 command port).
6. The maximum number of PCs when using 1 data port of PDS = Available sockets/(1 data port + 1 command port).
7. IO Port is 9999. (Only support for the module which has the DI/O.)
8. The web uses the TCP port 80. (It can be disabled.)
9. The telnet uses the TCP port 23. (It can be disabled)
10. The UDP search function will occupy one socket.
    - UDP = 0 → Doesn't support UDP search
    - UDP = 1 → Support UDP search and always occupy one socket
    - UDP = 2 → Support UDP search but while has the connection in that UDP search will be stopped.

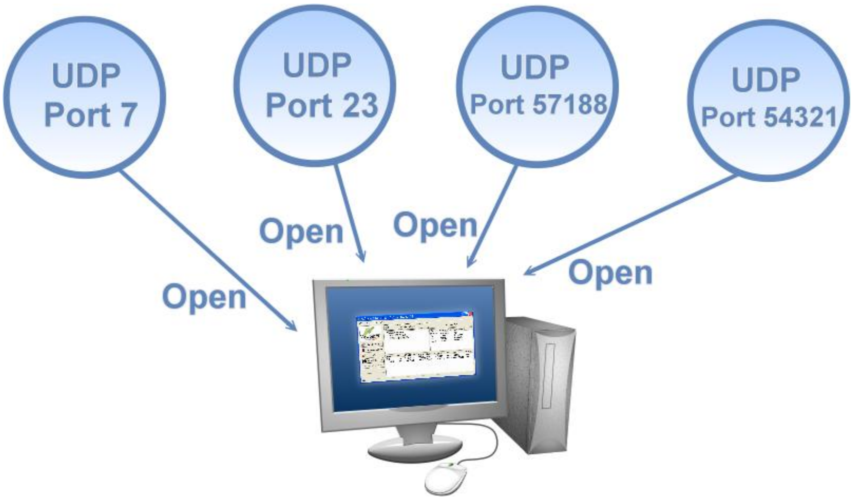
10. Can I search or connect to PDS when my PC’s IP address is not in the IP filter list of PDS? How can I solve it?

A. No, you cannot search or connect to PDS when the PC’s IP address is not in the IP filter list of PDS. You can solve the problem by:

1. Add your PC’s IP address into the IP filter table of PDS by using console command “IPFILTER” in section 9.4.1.
2. Or change your PC’s IP address to one of the IP addresses listed in IP filter table.
3. Or disable the IP filter function of the PDS by using console command. Refer to section 9.4.1.
4. Or clear all configuration setting on PDS by using “config=RESET” command on section 9.4.22. It also clears the IP filter table, password, alias... setting. You have to reboot the PDS for loading new configuration.

11. Why cannot computer ping or search the PDS series module?

The computer can make a communication with the module through some specific ports. Please confirm with your network administrator that UDP Port 7, Port 23, Port 57188 and Port 54321 can't be denying by network device.



About the detailed information of TCP/UDP port refers to following table:

- **TCP Port:**

Port Number	Description
80	HTTP (HyperText Transport Protocol)
999	DCON Port
10000	Command Port
10001	Serial Port for COM1
10002	Serial Port for COM2
10003	Serial Port for COM3

- **UDP Port:**

Port Number	Description
7	Echo (Ping)
23	Command Port
57188	Request of UDP search
54321	Response of UDP search

12. What is the difference in number of socket between tGW-700, 7188E2-MTCP and PPDS-700-MTCP?

7188E2-MTCP		tGW-700		PPDS-700-MTCP	
Model	Available Sockets	Model	Available Sockets	Model	Available Sockets
I-7188E2(D)-MTCP	29	tGW-712	4	PPDS-712(D)-MTCP	27
The 7188E2-MTCP series module has 32 sockets (includes some reserved listening sockets). So the number of Available Sockets is equal to the Max. Sockets (32) subtract Listening Sockets. For detailed available sockets information of 7188E2-MTCP series, refer to of FAQ03 of 7188e series. http://www.icpdas.com/faq/7188e/hardware/003.htm		tGW-722	8	PPDS-715(D)-MTCP	27
		tGW-732	12	PPDS-720(D)-MTCP	26
		tGW-715	4	PPDS-721(D)-MTCP	25
		tGW-725	8	PPDS-732(D)-MTCP	24
		tGW-735	12	PPDS-734(D)-MTCP	24
		tGW-718	4	PPDS-742(D)-MTCP	24
		tGW-724	8	PPDS-743(D)-MTCP	23
		tGW-734	12	PPDS-752(D)-MTCP	23
		The tGW-700 series module has 4 sockets in every serial COM port. So the number of Available Sockets is equal to the 4 Sockets by multiplied number of serial COM port.		PPDS-755(D)-MTCP	23
				PPDS-762(D)-MTCP	22
				PPDS-782(D)-MTCP	20
				The PPDS series module has 32 sockets (includes some reserved listening sockets). So the number of Available Sockets is equal to the Max. Sockets (32) subtract Listening Sockets. For detailed available sockets information of PPDS-700-MTCP, refer to of FAQ09.	